MICRO & PERSONAL computer

8

Lire 2500

SISTEMI APPLICAZIONI PROGRAMMI PERIFERICHE

m&p computer novembre-dicembre 1980 numero 8 - Spedizione in abbonamento postale gruppo III 70%

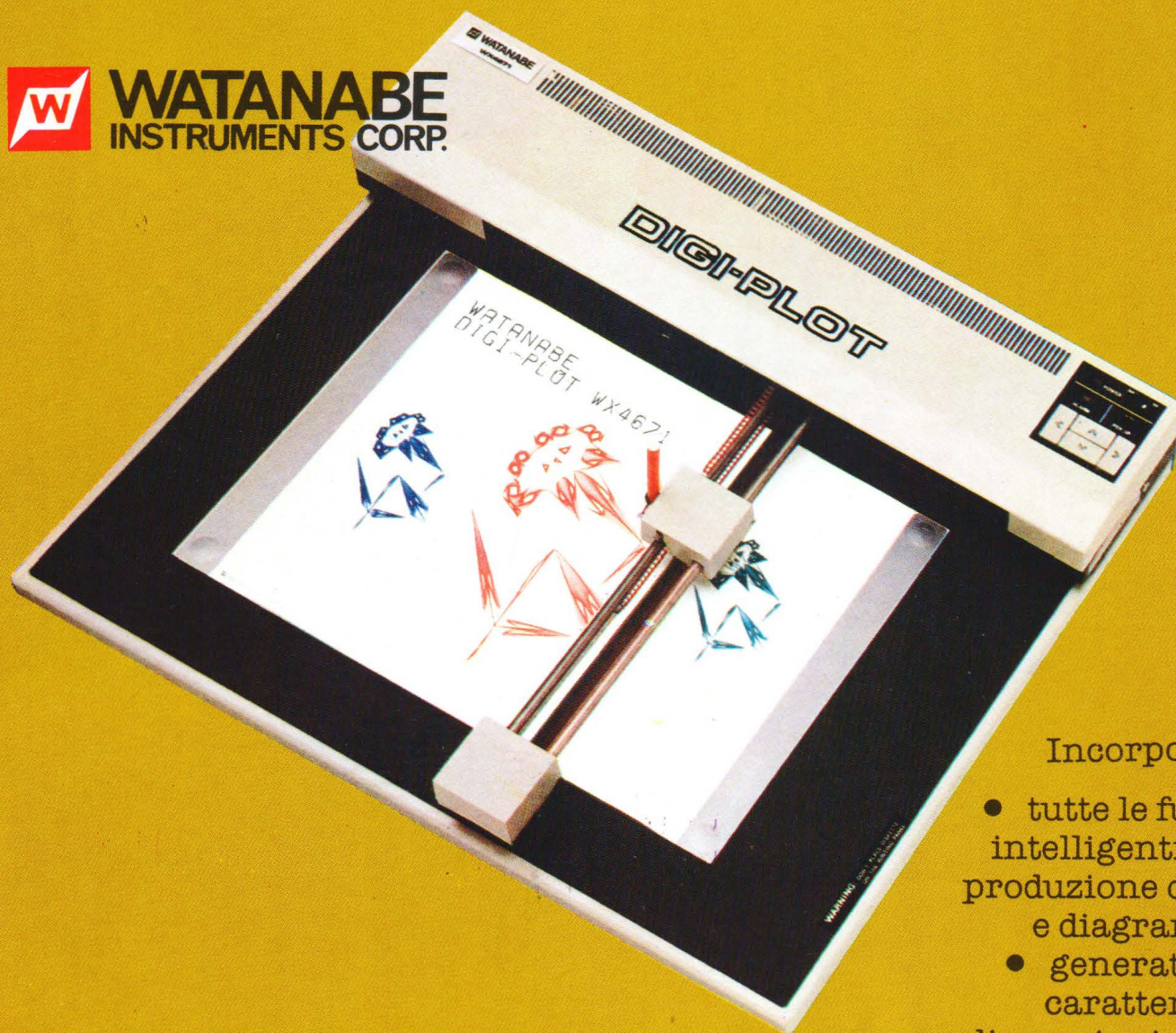
IN PROVA: CBM 8032
NANOCOMPUTER
EPROM PROGRAMMER PER APPLE

Il LISP a portata di PET
La divisione in sillabe
Crittografia computerizzata
Il micromputer nasce dal BUS
Il software dei lettori

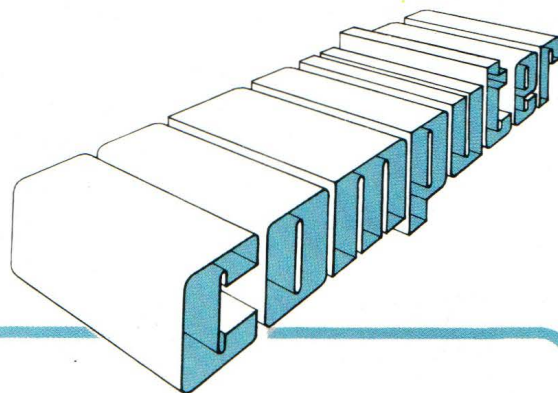
GUIDA MERCATO: i prezzi di personal computer
stampanti
calcolatrici programmabili
schede microcomputer

IL PLOTTER INTELLIGENTE ED ECONOMICO PER IL VOSTRO PERSONAL COMPUTER

Incorpora:

- tutte le funzioni intelligenti per la produzione di grafici e diagrammi
- generatore di caratteri di dimensioni variabili per lettere, numeri e simboli
- possibilità di funzionamento in «printer mode»
- interfaccia parallela a 7 bit, codice ASCII per il collegamento sull'uscita «printer» del microcomputer

ECTA

Via Giacosa, 3 - 20127 Milano
Tel. 2895978 - 2829907
Telex 313039 ECTAMI

diga

In una stupenda valigetta rossa, un eccezionale maestro per insegnarvi tutto sullo Z 80.

Nanocomputer

pag. 42

La divisione in sillabe

ovvero, come insegnare al vostro computer ad «andare a capo» rispettando le regole della grammatica italiana.

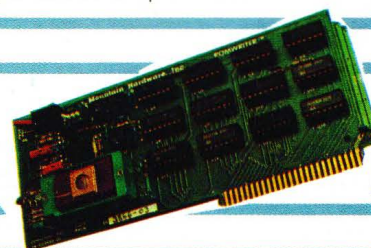
pag. 31

Il LISP a portata di PET

Per imparare un linguaggio può bastare un libro, ma solo per la teoria. Per la pratica, ci vuole un computer sottomano. E l'interessantissimo LISP è disponibile anche per il PET.

pag. 22

EPROM programmer per Apple

Trasferire un programma su EPROM è comodo e facile con questo EPROM Programmer della Mountain Hardware: noi ci abbiamo provato con il renumber dell'Apple.

pag. 48

Crittografia

Insegnate al vostro computer a custodire gelosamente qualsiasi segreto...

pag. 55

Il PET è diventato grande: un video da 80 colonne ed una nuova versione del BASIC ne hanno ampliato notevolmente le possibilità... ma non (forse) il prezzo.

CBM 8032

pag. 36

Entrare in possesso dell'8032, il nuovo Commodore a 80 colonne impropriamente noto come il PET grande, non è stato facile. Alla fine, non senza suscitare polemiche ancor prima dell'uscita della rivista, ne abbiamo avuto un esemplare importato direttamente dall'America grazie all'intraprendenza della Deniel's di Torino. Acquistare un Nanocomputer è invece molto più facile: ha persino il manico. Stiamo scherzando, ovviamente, ma la cura posta nella confezione è solo la punta dell'iceberg della cura, dell'amore, della professionalità poste dalla SGS nel progetto e nella realizzazione di quell'eccezionale sistema didattico che si chiama Nanocomputer.

Foto: Gianfranco Machelli
Grafica: Gaetano Giaquinto

ANNUARIO

Avrà avuto si e no quindicianni. All'apertura dell'EDP USA è arrivato al nostro stand, si è piazzato davanti ad un HP-85 e non l'ha più mollato. Dopo qualche ora mi dice: «come si fa a generare un numero casuale compreso tra 3 e 5? Scusi, sa, ma questa è la prima volta che faccio un programma». Due giorni dopo il suo programma in BASIC, un «Master Mind», girava regolarmente. Questo tanto per chiarire il tipo.

Mentre il nostro amico quindicenne scriveva il suo primo programma in BASIC, Silvano Fraticelli si aggirava per la mostra chiedendo il prezzo di questo e di quello per gli ultimi ritocchi alla guida mercato dell'ANNUARIO di m&p COMPUTER. A un certo punto torna trafelato al nostro stand e mi annuncia: «Alla... non mi vogliono dare i prezzi aggiornati, non vogliono che li pubblichiamo». Io, più presuntuoso del solito per le sempre più numerose manifestazioni di consenso che nel frattempo stanno giungendo, sentenzio: «Niente prezzi, niente prodotti nella guida mercato».

A questo punto il nostro amico quindicenne, un metro più in là solleva la testa un attimo ed esclama: «Basta, è ora di finirla con questa storia che non vogliono dire i prezzi, non c'è nulla di più indisponente che vedere una cosa e non sapere quanto costa». Grande ed elementare verità alla quale, a parte qualche caso isolato, si è ormai adeguata la stragrande maggioranza degli operatori commerciali del settore personal computer che, ormai, non fanno più resistenza a seguirci nella nostra politica di massima informazione sui prezzi.

E, giungo al punto, siamo dunque orgogliosi di presentare l'ovvio (per noi) sottoprodotto di questa altrettanto ovvia (sempre secondo noi) politica: il primo ANNUARIO di m&p COMPUTER. Ehi, attenzione, solo la nostra modestia ci induce a definirlo sottoprodotto, ma in effetti si tratta di una cosetta niente male: 152 pagine comprendenti guida-mercato di personal computer, periferiche e accessori, calcolatrici programmabili, schede microcomputer, software per personal computer, software per calcolatrici programmabili in SOA e RPN, e, in più, un dizionario dei termini difficili, una introduzione al mondo del calcolo elettronico, e molti programmi in BASIC, RPN, SOA. Quando sarà in edicola? 7-10 giorni dopo l'uscita di questo numero. E il prezzo? 3.000 Lire. «Aoh, ve state affa' pubblicità». Ebbene si, lo confesso.

Paolo Nuti

micro & personal COMPUTER

Anno II - n. 8 - mensile - Novembre/Dicembre 1980 - Lire 2.500
Spedizione in abbonamento postale gruppo III 70%

Direttore: Paolo Nuti
Coordinatore: Marco Marinacci
Grafica e impaginazione: Gaetano Giaquinto e Diana Santosuosso
Segretaria di redazione: Giovanna Molinari
Fotografia: Gianfranco Machelli, Luciano Marinelli
Direttore responsabile: Gianfranco M. Binari
Direzione editoriale: Gianfranco M. Binari e Daniel Caimi
Hanno collaborato: Bo Arnklit, Mauro Boscarol, Silvano Fraticelli, Paolo Galassetti, Pietro Hasenmajer, Marialba Italia, Filippo Merelli, Lorenzo Mezzalira, Gianni Romeo, Gualtiero Rudella, Pierluigi Panunzi, Pietro Tasso.

m&p COMPUTER è una pubblicazione del Gruppo Editoriale Suono s.r.l. - Via del Casaletto, 380 - 00151 Roma, telefono (06) 538.041 (6 linee con ricerca automatica), telex: 614321 Edsuon I - Registrazione del Tribunale di Roma n. 13897 del 30-4-1971 - Sped. abb. post. gr. III 70% - Manoscritti e foto originali, anche se non pubblicati, non si restituiscono - Concessionaria per la pubblicità: Publisuono s.r.l., Via del Casaletto, 380 - 00151 Roma, telefono 538.041 (6 linee con ricerca automatica).

Servizio abbonamenti e arretrati: Via Giovanna Gazzoni n. 42 - 00133 Roma - tel. 2692848-2692809 - Abbonamento a 12 numeri: Italia L. 25.000, estero Europa L. 29.000, Americhe, Giappone, etc. L. 45.000 - C/c postale n. 774018 intestato a: Gruppo Editoriale Suono - Via del Casaletto 380 - 00151 Roma - Arretrati: 1 copia L. 3.000 - C/c postale n. 774018 intestato a: Gruppo Editoriale Suono - Via del Casaletto 380 - 00151 Roma.

Composizione: Velox, Via Tiburtina 196, Roma - Stampa: Romagraf, via Rina Monti 30, Roma - Concessionaria per la distribuzione: Parrini & C s.r.l., aderente adn, Piazza Indipendenza 11/B Roma, tel (06) 4992, Via Termopili 68, Milano, tel. 2896471.

ASS. USPI

POSTACOMPUTER

Postacomputer pubblica le lettere ritenute di interesse generale.
Il nostro indirizzo è: m&p COMPUTER - Postacomputer - Via del Casaletto, 380 - 00151 Roma.
Preghiamo i lettori di non richiedere risposte personali, né tanto meno inviare francobolli, buste affrancate, telex di sollecito etc. Ci è materialmente impossibile rispondere a tutti.
Tutte le lettere ricevute vengono lette con la massima attenzione e nel definire la linea della rivista teniamo conto dei suggerimenti e delle richieste dei lettori.
Alle lettere di interesse generale rispondiamo sulla rivista compatibilmente con lo spazio a disposizione.

COSTUME

Corsi di specializzazione per programmatori?

Mi rivolgo a voi, essenzialmente, per richiedervi un consiglio sulle possibilità di impiego nel campo della tecnica e dell'utilizzazione dei computer. Sinteticamente vi espongo la mia richiesta: essendo un giovane diciottenne che nell'anno in corso conseguirà un diploma di Perito Elettronico, in quel di Como, desidererei avere informazioni più dettagliate a proposito dei vari corsi di specializzazione per programmatori.
Non posso nascondervi la mia notevole attrazione verso i computer e le calcolatrici programmabili che, peraltro, ho già «usato», sia gli uni che le altre.
Sono abbonato alla vostra interessantissima rivista sin dal primo numero e colgo l'occasione per ringraziarvi dei vostri preziosissimi consigli riguardo le calcolatrici; insomma, sono rimasto entusiasta delle prestazioni che possono offrire le calcolatrici programmabili come la mia, forse la più piccola, ma, per me, versatilissima, la TI-57 che avete così ben presentato nel vs. 1° numero. In conclusione vi chiedo di farmi avere alcuni recapiti di sedi per corsi di specializzazione, nelle vicinanze di Como o Milano, in modo che io possa avere dei punti di riferimento per il mio avvenire.

Achille Montini
Cantù

Non ci risulta che vi siano corsi non universitari di specializzazione in informatica statali o comunque gestiti da una struttura pubblica. Vi sono, viceversa, molti corsi di specializzazione, la cui durata va da un giorno a un paio di settimane, organizzati dai costruttori di questo o quel sistema, il cui costo è generalmente molto elevato e che si rivolgono o al personale di clienti già acquisiti o a possibili nuovi clienti.
Accanto ai corsi organizzati dai costruttori di sistemi vi sono altri corsi di specializzazione particolarmente seri, e citiamo per tutti quelli organizzati dalla scuola di elettronica (Via Vittor Pisani, 22 — Milano — Tel. 6572815) che peraltro per programmi, struttura dei corsi e costi si rivolgono essenzialmente a professionisti o comunque dipendenti di aziende.
Ma forse lei si riferiva a qualcuno di quei «Corsi per perforatrici e programmatori IBM» etc. etc. che abbondano tanto a Milano che a Roma, ma dei quali, a parte la martellante pubblicità murale e autofilotranviaria non abbiamo alcuna esperienza, né in bene, né in male.
Pubblichiamo la sua lettera nella speranza che qualcuno dei nostri lettori, alunno e ex alunno di detti corsi, abbia voglia di comunicarci qualche impressione sulla sua esperienza.

MERCATO

Interfacciamento 8080 e Bus MMS-8

Sono uno studente del 5° anno di uno dei pochi istituti per l'informatica della Campania, essendo un appassionato di questa materia compro assiduamente la vs. rivista e colgo l'occasione per esprimervi le mie più sincere congratulazioni ed auguri per un buon proseguio.
Il motivo di questa mia è ben preciso: essendo in possesso di un 8080A, e di un pò di materiale SOFTWARE per questo microprocessore, chiederei gentilmente se mi potreste citare dei testi che mi guidassero nell'interfacciamento di questo chip, e possibilmente sapere dove potrei reperire il BUS descritto eccellentemente dal vostro collaboratore Lorenzo Mezzalira.
Approfitto per lanciare un messaggio agli amici del «COMPUTER»: posso dare utili informazioni sul personal computer SUPERBRAIN e sul suo sistema operativo.

Lorenzo Tornatore
Via Calata Capodichino, 243
Napoli

Sull'interfacciamento dell'8080 non manca certo un'ampia lette-

Riceviamo e volentieri pubblichiamo: quale ruolo per l'Informatica

Informatica: neologismo originato dalla fusione delle due parole «informazione» «automatica» significa trattamento automatico dell'informazione.
L'Informatica è entrata di forza nella nostra vita di ogni giorno; fa parte della cultura della società del 2000, nel senso che è fondamentale alla sua sopravvivenza nella sua struttura attuale, buona o cattiva che sia, ed è uno strumento teorico indispensabile per comprendere la realtà e modificarla.
È proprio relativamente al fatto «Cultura» che la situazione, per lo meno in Italia, è sconsolante. La nostra Cultura Ufficiale sembra essere solo la cultura così detta umanistica.
Anche la Scuola, che istituzionalmente ha il compito di trasmettere la cultura alle nuove generazioni, cioè riprodurre quel bagaglio di conoscenze, strumenti teorici e pratici, formazione intellettuale che assicurino la sopravvivenza e l'evoluzione della Società, ha di fatto relegato le scienze e le tecniche in un canto, non solo limitandosi a trasmettere unicamente le scienze «tradizionali» nel modo che tutti conosciamo, e trascurando realtà attuali quali Sociologia, Statistica, Ricerca Operativa, Informatica, ma soprattutto, ed è questo il fatto grave, isolandole, impedendo qualsiasi interazione con il Sapere Vero. Lo stato della ricerca scientifica testimonia bene il trionfo di questa mentalità a tutti i livelli, pochi sono coloro che sentono il problema in tutta la sua gravità e quei pochi sono tutt'altro che agevolati nella loro opera di diffusione di una corretta presa di coscienza da parte di tutti.
Le scienze dell'informazione in questo quadro non sono certo privilegiate: nella scuola di massa le scienze dell'informazione non esistono, tutt'al più negli istituti ad indirizzo tecnico o professionale vengono ridotte al puro ruolo di tecniche, negandone di fatto il ruolo culturale.
Che il problema non sia sentito a nessun livello è evidente anche dalla mancanza nella nostra legislazione di qualsiasi norma che regoli l'uso dell'informazione e tuteli il cittadino: l'Informazione è Potere.
Leggi di questo genere esistono da molto tempo in tutti i paesi progrediti, in compenso mancano del tutto nei paesi retti da regimi totalitari.
L'informazione è la linfa di tutte le attività umane: è importante capire bene che cosa essa sia, ed è un concetto veramente difficile da definire proprio per la vastità di significati, è difficile da definire in quanto dire solo che cosa è e che cosa non è risulta alla fine estremamente riduttivo, poiché trascura il suo aspetto fondamentale di azione finalizzata.
«Una informazione in sé non è niente. Non è un'informazione se nessuno ne viene informato o se l'informazione è inutilizzabile da colui che l'ha ricevuta». (1)
Si diceva su queste pagine che gli analfabeti del futuro saranno coloro che non conosceranno almeno un linguaggio di programmazione: è vero ma non basta, e mi spiego con un esempio.
Un marziano ipotetico capitato sulla terra, che in un tempo brevissimo fosse in grado di imparare a perfezionare la nostra lingua trarrebbe dalla lettura di un quotidiano la stessa utilità di un analfabeta che non è in grado di leggerlo, in quanto gli mancherebbe comunque quel retroterra culturale che gli permetterebbe di «estrarre» i significati veri dalle parole.
Certo, come in tutte le cose, la padronanza del linguaggio è il primo requisito per affrontare in modo proficuo qualsiasi realtà.
La diffusione dei computer oggi ci dà la possibilità di accostarci con facilità all'informatica ed alle scienze dell'informazione con le conoscenze tecniche necessarie.
E non bisogna temere di inaridirsi, ridursi a macchine, a robot: se la programmazione oggi può essere un fatto meramente tecnico, l'informazione è economia, sociologia, musica, antropologia, politica. Oggi l'imperativo è questo, l'alternativa è l'emarginazione.

Giovanni Cornara

(1) Bonsack: «Il concetto di informazione nella scienza contemporanea».

ratura! A parte i manuali originali della Intel che può reperire presso il distributore italiano (Eledra 3S, Milano) esiste una specie di breviario sul quale si è preparata una intera generazione di programmatori dell'8080: si chiama Bugbook III; la traduzione italiana è reperibile presso tutte le sedi GBC.
Un sistema utilizzante il bus MMS-8 descritto negli articoli del Professor Mezzalira, viene costruito in Italia da L'Emmeci, via Porpora, 132 — Milano.
Per permettere ai lettori interessati di mettersi in contatto con lei, pubblichiamo il suo indirizzo per esteso.

Cancellare con la stampante?

Per lavoro ho bisogno di una stampante, abbinata ad una tastiera ed ad una memoria possibilmente in floppy-disc, che possa stamparmi una serie di indirizzi su un dato modulo continuo. Il problema è dato dal fatto che a me interesserebbe una stampante che possa, dopo aver terminato di stampare il primo indirizzo, ritornare tre o quattro righe in alto e ricominciare. Mi necessita anche la possibilità di cancellare un intero indirizzo senza danneggiare gli altri o aggiungerne.
Ovvero una stampante in grado di far scorrere la carta nei due sensi un poco similmente alle normali macchine da scrivere per scrivere gli esponenti ecc.
Pur seguendo m&p COMPUTER dal n. 2, non ho ancora definito il modello che può interessarmi. A questo punto posso da voi ottenere un suggerimento?

Livio Ballarini
Riccione

Molte stampanti consentono di effettuare il salto di riga all'indietro (back space), sia spostando la carta di una intera riga, sia di mezza riga (p.e. Centronics 737), sia di incrementi minori (p.e. Qume, HP 9871).
Nessuna stampante a noi nota offre invece la possibilità di cancellare parte di quello che ha scritto. Né riusciamo a comprendere il suo interesse per questa funzione in una stampante da adibire alla stampa di indirizzi.
Il bello di svolgere il lavoro servendosi di un computer è proprio nella possibilità di correggere gli errori sul video! Dopo la fase di correzione il computer riordina i dati da stampare nella sequenza desiderata (in gergo fa un «Sort») e poi stampa il tutto in rapida successione. Il «tasto di cancellazione», così comodo nella scrittura manuale, diventa, con il computer, del tutto superfluo.

TECNICA

Quante periferiche per una 41-C?

Sono uno studente di ingegneria ed un appassionato di micro e mini computer. Ho trovato la vostra rivista molto interessante ed approvo pienamente l'impostazione.
Vi ho scritto per porre alcuni quesiti che spero possano essere d'interesse generale.
Ho letto sul n° 2 della rivista la prova dell'HP-41C e non ho ben chiare le possibilità di interfacciamento; per esempio se ho un programma in macchina occupante tutti i quattro moduli di espansione di memoria, è possibile utilizzare la stampante o qualcos'altro? Oppure, una volta occupati i quattro slots, non si può aggiungere niente altro?

Germano Longo
Pozzuoli

Ahimé è così, se il programma occupa 4 moduli di memoria è impossibile inserire altre periferiche. Se per esempio vogliamo collegare la stampante o il lettore, non si possono inserire più di 3 moduli di memoria; volendo collegare sia la stampante che il lettore, il numero massimo dei moduli inseribili è 2. Tuttavia è possibile, se si ha necessità di usare lettore di schede e stampante, inserire ugualmente 3 moduli di memoria inserendo il lettore nel quarto slot e eseguendo con esso le operazioni necessarie, dopodiché lo si può togliere per inserire la stampante; nel caso dovesse servire di nuovo il lettore, si toglie la stampante e si reinserisce il lettore (ricordando ogni volta di spegnere la macchina). In pratica si usa un solo slot per collegare due periferiche che non hanno mai necessità di lavorare contemporaneamente.
Tra breve vi sarà una ulteriore possibilità: secondo attendibili indiscrezioni, verso gennaio dovrebbero entrare in distribuzione dei nuovi moduli di memoria a quadrupla densità con i quali il suo problema sarà risolto drasticamente.

HP 41-C: problemi di plottaggio

Vi scrivo poiché penso siate gli unici a poter dare una risposta ai miei problemi. Sono già da qualche anno nel mondo delle calcolatrici programmabili, (come utente) prima come possesore di una Texas SR-52 ed ora di una HP

41C con lettore di schede, stampante termica e due moduli di memoria; i problemi sono questi:
1) . . . .
2) Con il mio sistema è possibile fare tracciare il grafico di una funzione matematica y=f(x) alla stampante richiamando il programma «PRPLOT». Ciò che a me servirebbe è la possibilità di tracciare i grafici di più funzioni simultaneamente, quindi vorrei sapere se è possibile avere, pagando naturalmente, del software riguardante tale problema.
3) È possibile costruire un'interfaccia per collegare alla mia calcolatrice un plotter? Ad esempio il 7225 A della HP.
Rigraziandovi anticipatamente per l'attenzione che vorrete darmi, porgo distinti saluti.

Stefano Gragnani
Lucca

Sicuramente avrà notato che la funzione «PRPLOT» è, in sostanza, un progrmma residente in ROM (nella stampante) e creato per tracciare grafici di una sola funzione alla volta. Il grafico viene costruito per punti, distanti tra loro di un valore pari all'interlinea, usando altre due funzioni contenute nella ROM della stampante: «PRAXIS» e «REGPLOT». «PRAXIS» traccia l'asse Y e segna su di esso tre punti: Ymax, Ymin e l'origine degli assi; «REGPLOT», ad ogni incremento Δx dell'ascissa, stampa un simbolo in corrispondenza del valore f(x).
Venendo al suo problema, per tracciare contemporanemente due funzioni si possono adottare principalmente due soluzioni. La prima, e anche la più semplice, consiste nel far stampare il valore di f(x) poi, alla riga successiva, il valore g(x+Δx) e poi ancora f(x+2Δx) e così via, alternando tra loro le due funzioni y=f(x) e y=g(x), per esempio con un flag che viene ad ogni interazione acceso o spento. Ovviamente questo metodo costringerebbe l'operatore a scegliere la stessa scala sia per l'asse Y che per l'asse X delle due funzioni; inoltre tra un punto e l'altro del tracciato relativo alla stessa curva si avrebbe una distanza pari a due volte l'interlinea, rendendo così la lettura del grafico poco immediata.
La seconda soluzione, quella che ci sembra la migiore, è anche la più difficile da realizzare. Si tratta di far tracciare contemporaneamente, sulla stessa riga, i due punti f(x) e g(x); in questo caso non è possibile usare la funzione «PRPLOT» e neanche la «REGPLOT», perché è necessario costruire interamente una routine in grado di formare là riga contenente i due punti, ed anche un tratto dell'asse X lungo ΔX, avvalendosi delle funzioni «ACCOL» e «SKPCOL».
Il problema, comunque, è molto interessante e verrà trattato più ampiamente in seguito.
Attualmente non esiste alcuna possibilità di interfacciare la 41C con un plotter.

Errore Race

Sono abbonato alla vs. rivista e la seguo con interesse sempre maggiore: vi faccio i miei complimenti e vi porgo i miei migliori auguri.
Quando decisi di abbonarmi (dopo l'uscita del 2° numero) ero interessato soltanto ai microcomputer: ora mi sento un po' attratto dai «PERSONAL»; mi auguro, comunque, che questa rivista diventi anche un veicolo di scambio di esperienze di programmazione: credo che ne trarremmo tutti maggior soddisfazione!
Come proprietario di una HP-41C mi sono dilettato a giocare col vs programma (cfr. n° 6) «RACE», che ho trovato molto divertente.
Da un esame eseguito sul vs programma (ben fatto sintatticamente ma con un piccolo difetto peraltro facilmente eliminabile) ho «scoperto» (cfr. LABEL Ø6) che con opportuna velocità sarebbe possibile trattare una doppia curva come se fosse un rettilineo!!
Mi spiego subito: il tracciato è suddiviso in 61 tratti le cui caratteristiche (orientamento e curvatura) sono memorizzate in altrettanti registri; con velocità (numero intero) n si percorrono n tratti nell'unità di tempo (impiegando, cioè, soltanto tempo 1); il controllo della posizione dell'auto sulla pista viene eseguito soltanto al termine dell'operazione condotta nell'unità di tempo (se si sta affrontando una curva con velocità n, la posizione dell'auto viene controllata soltanto all'nesimo tratto): da queste premesse si ricava subito che secondo il vostro programma un'auto che affronti con velocità n (o maggiore) una doppia curva definita con n tratti può rimanere in strada se i modificatori della curva si compensano tra di loro. L'esempio che segue chiarirà meglio il concetto.
Costruiamo un «circuito aperto» così definito: CLRG, 1 STO 38 STO 41, CHS, STO 42 STO 43, 2 STO 39 STO 4Ø, consistente, in sostanza, in un rettilineo con una sola doppia curva: premendo ripetutamente «AC» fin dall'inizio e premendo AC (o non toccando alcun tasto) anche all'approssimarsi della curva come se volessimo andare diritti, l'auto non esce di strada!

Vediamo perché: l'auto si trova al centro della strada (posizione 6) e procede a velocità 6 (se all'approssimarsi della curva non abbiamo premuto alcun tasto: se abbiamo premuto AC, avrà velocità 7): dopo il primo tratto (R38) l'auto assume posizione 7, dopo il secondo (R39) pos. 9, dopo il 3° (R4Ø) pos. 11 (e qui sarebbe uscita di strada!), dopo il 4° pos. 12, dopo il 5° pos. 11 e dopo il 6° pos. 10, ossia l'auto rientra in strada come nulla fosse: poiché il controllo avviene solo in questo momento, la curva risulta superata in modo corretto! I modificatori dei registri R42 ed R43 hanno compensato i modificatori precedenti e la velocità (almeno 6 per superare in un solo colpo tutti e 6 i tratti, tutti e sei i registri della curva) ha consentito che intervenissero anche i registri R42 e R43. Naturalmente basta porre un'istruzione XEQ Ø5 tra le istruzioni 98 ST+15 e 99 DSE ØØ per mettere tutto a posto: l'elaborazione sarà leggermente più lunga, ma più corretta.

Fabio Lusiani
Primolano (VI)

In effetti, col programma originale si poteva «gabbare» la 41C, sia pure rischiando di uscire di strada. La modifica del Sig. Lusiani rende senz'altro più corretto il procedimento sebbene a discapito della velocità di elaborazione. Se si effettua tale modifica, il passo 3Ø XEQ Ø5 diventa superfluo e va eliminato.

Ancora Race

Spett. m&p COMPUTER,
vi scrivo a proposito del programma (eccezionale) RACE pubblicato sul numero 6 della vostra riuscitissima rivista. Ho riscontrato un errore che potrebbe essere sfuggito ai compilatori del listato in quanto non sempre si ripete.
Se giunti verso la fine del tracciato accade che nella subroutine 06 il decrementatore di velocità (R ØØ) arrivi a zero, e quindi rinvii alla subroutine Ø4, quando R 14 è stato incrementato sino a 76, la suddetta subroutine dopo tre passi si arresta visualizzando «NONEXISTENT». Ciò perché al passo 21 RCL IND X verrebbe richiamato indirettamente il registro che non esiste. Questo inconveniente si verifica ovviamente solo quando l'auto è ferma in corrispondenza del R 76 e non quando il «traguardo» viene oltrepassato, diciamo, in velocità.
Per risolvere il problema secondo me si potrebbe modificare il programma come segue:

```
20  +
21  77
22  X≷Y
23  X=Y?
24  GTO Ø6
25  RCL IND X
```

Sperando di aver fatto cosa gradita a qualcuno vi saluto invitandovi a continuare.

Marcello Priaglio
Torino

OK! Così va tutto bene; però ci conceda una piccola pignoleria: sostituire il GTO Ø6 del nuovo passo 24 da lei proposto, con un GTO Ø7. Questo perché quando la macchina salta alla LBL Ø6, e questo avviene se R14+1=77 (cioè R14=76), ai passi 91 e 92 (del programma originale) tale registro verrà incrementato di una unità, esso diverrà quindi R14=77 con conseguente esito positivo del test di cui al passo 95 e quindi salto alla LBL Ø7. Tanto valeva perciò che la macchina saltasse direttamente dal passo 24 alla LBL Ø7, abbreviando così i tempi d'esecuzione. Comunque la nostra è una precisazione «di forma», perché il programmma gira benissimo anche con il GTO Ø6 al passo 24, come proposto dal Signor Priaglio.

Errore Poker

Sono una studentessa in matematica, ed ho recentemente scelto un indirizzo orientato verso i calcolatori elettronici. Premetto che solo da poco (2 numeri) compro la vostra rivista, e quindi non mi sono ancora formata un giudizio; apprezzo però molto l'impostazione da voi effettuata, che mi permette di poter disporre di un mezzo di informazione sui m&p COMPUTER e nello stesso tempo mi permette di ampliare le mie conoscenze in materia, che purtroppo non sono attualmente molte. Ma veniamo al dunque: sono in possesso di una calcolatrice programmabile HP 34C, per cui, quando sul numero 5 di m&p è uscito un articolo su di essa, l'ho accolto con «summa» gioia. Ho anche cercato di eseguire il programma da voi proposto a pag. 36, ma dopo due tentativi inutili di farlo girare così com'era, ho notato che probabilmente alcune istruzioni del vostro listing sono state stampate in modo errato: precisamente le linee ØØ5, Ø13, Ø17, Ø96 dove, secono me, al posto del numero «1» dovrebbe comparire «I»; cioé:

```
ØØ5 STO fI
Ø13 STO fI
Ø17 STO fI
Ø96 STO fI
```

Con le sostituzioni da me eseguite il programma gira perfettamete.

Ornella Corridori
Roma

Ci fa piacere osservare che il mondo del «micro & personal» sia abitato anche dal gentil sesso, sebbene la cosa non ci stupisca affatto. Per quanto riguarda l'errata corrige, non si sembra il caso di aggiungere altro, se non le nostre scuse a chi ha tentato invano di far girare il programma. A tale proposito facciamo presente che i programmi per HP 41C e HP 67/97 pubblicati, sono esenti da errori di trascrizione poiché di essi viene riprodotto direttamete il listato stampato dalla macchina.

Errata corrige
Rettifichiamo l'elenco dei rivenditori SWTPC 6809 apparso sull'inserzione pubblicitaria HOMIC sul numero precedente di m&p COMPUTER.

Distributori Homic:

DIGITRONIC S.r.l.
via Provinciale, 46
22038 Tavernerio (CO)
tel. 031/427076

HOME DATA SYSTEM
via Vercellotto, 134/b
13014 Cossato (VC)
tel. 015/93770

K-BYTES
di Alberto Capini
via XX Settembre, 20/12
16121 Genova
tel. 010/592636

CO.R.EL. - FRIULI
via Mercatovecchio, 28
33100 Udine
tel. 0432/291466

E.E.C.
di Ing. Santi Currò
via G. La Farina, 40
98100 Messina
tel. 090/2924164

COMPUTER SYSTEM
di Sergio Salgaro
via Fama, 15
37121 Verona
tel. 045/23581

VIDEODATA
Borgo S.ta Chiara, 12
43100 Parma
tel. 0521/33989

SACAT
via Duco, 34
25100 Brescia
tel. 030/381337

Varie TI-59

Sono un vostro affezionato lettore e seguo con grandissimo interesse la vostra rubrica «Software S.O.A.» perché posseggo un calcolatore Texas Instruments T-59. Vi scrivo per proporvi anche un mio programma, ma prima di descriverlo vorrei che voi mi spiegaste alcuni strati comportamenti della TI-59:
1) cosa significa l'oscuramento del display, preceduto da una breve visualizzazione del numero del blocco di memoria, dopo la registrazione di una scheda? (So che per rendere visibile il numero del blocco di memoria, e quindi riaccendere il display, è sufficiente premere R/S, ma il programma è stato registrato correttamente o è un caso che ciò avvenga?);
2) perché nella seguente sequenza che sarebbe dovuta servire per risparmiare 3 passi di programma nella stampa di una «E2» a capo di una delle 4 colonne di cui è composto il nastro della stampante PC 100 C la calcolatrice «ruba» una unità alla cifra-risultato esatto (170000000)?
17 x 8 INV 2nd log = 2nd op 04 (oppure 01, 02, 03).
Infine vorrei chiedervi se è conveniente usare qualche volta la calcolatrice accoppiata alla stampante senza inserire le batterie sotto carica nell'apposito vano, per permettere che esse, come è consigliato nelle istruzioni, possano scaricarsi «quasi del tutto».

1) Che il display rimanga oscurato alla fine della lettura di una scheda è strano, rispettò alle altre calcolatrici in cui si riaccende spontaneamente. Nella mia calcolatrice (vedi risposta a Mancinelli sul n° 4) rimane acceso il motorino di trascinamento e, se la scheda è stata effettivamente letta, per spegnerlo bisogna appunto premere R/S: se però la scheda non è stata letta, non si riesce a bloccare il motorino se non spegnendo la calcolatrice con consenseguenze immaginabili. Penso che nel suo e nel mio caso si tratti di un piccolo malfunzionamento del «micro-switch» azionato dalla scheda stessa e che fa partire il motorino.
2) In genere quando si usa la INV Log *bisogna stare attenti alle approssimazioni introdotte da questa funzione per i valori alti della variabile. Nel caso in questione premendo* 8 INV Log *sul display si legge il risultato corretto (100000000) ma premendo successivamente* Int *vediamo la parte intera del risultato «effettivo» che è stato poi arrotondato in base alle cifre decimali di scorta. Ora anche le operazioni di stampa (ad es.* O^P 0^4*) operando sulla parte intera del numero «effettivo» e non del numero arrotondato: in questo caso infatti si ha 9999999 che è errato per i nostri scopi. Per ovviare a questo inconveniente, subito dopo* INV Log *bisogna troncare le cifre di scorta e far sì che il risultato sia quello effettivamente visibile sul display. Ciò si ottiene con* EE INV EE *oppure, nel caso che il risultato debba essere sicuramente intero, con* D.MS *dal momento che entrambe operano solo sul dato visualizzato (vedi manuale a pg. D-2). In alcuni programmi del modulo Math/Utilities in casi simili viene appunto usato* D.MS*, la cui presenza è altrimenti inspiegabile.*
3) Non credo sia consigliabile. Basta togliere la spina della stampante e gli accumulatori si scaricheranno per conto proprio.

Pierluigi Panunzi

Homic il più grande centro italiano di microcomputer propone:

Commodore CBM*

Il potente microsistema per applicazioni gestionali e professionali. Unità centrale 32 K RAM - Doppia unità disco - Video terminale - Stampante veloce. Disponibili programmi di contabilità, fatturazione, magazzino e programmi personalizzati.

* Importatore esclusivo: Harden SpA Sospiro (Cremona)

Alla Homic trovi altri bei nomi: come Texas Instruments, Hewlett Packard, Commodore, Nascom; e i "personal" più avanzati, con diverse capacità di memoria, prezzi su misura, periferiche per tutti gli usi, supporti per programmazione e programmi personalizzati. E trovi assistenza. Nella scelta e dopo. Vuoi un "micro"? Vai in negozio e comperalo.

Milano - uffici: Piazza De Angeli 3 - Tel. 4695467/4696040
centro vendite: Galleria De Angeli 1 - Tel. 437058

Distributori Homic:

BONARDI PRIMO
24100 Bergamo
tel. 035/222151

E.D.S.
20145 Milano
tel. 02/4985326

I.S.S.
21047 Saronno (VA)
tel. 02/9609971

SELETRA
21049 Tradate (VA)
tel. 0331/843488

DIGITRONIC
22038 Tavernerio (CO)
tel. 031/427076

SACAT
25100 Brescia
tel. 030/381337

ERRE-PI-ERRE srl
27058 Voghera (PV)
tel. 0383/45831

I.C.S. srl
20129 Milano
tel. 02 745384

ELCOD sas
24011 Almè (BG)
tel. 035/542218

BETA SISTEMI sas
di Roveda Piero & C.
21053 Castellanza (VA)
tel. 0331/503991

ELETTRODATA
25100 Brescia
tel. 030/40896

LOTUS srl
20127 Milano
tel. 02/2592095

NOTIZIE APPLE

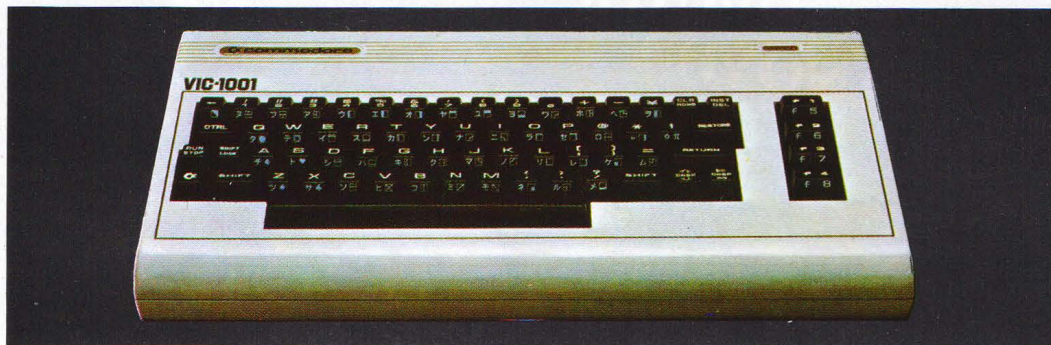
KEY PAD PER APPLE II

Se utilizzate frequentemente Dati Numerici, è disponibile un utile tastierino numerico di Styling Apple.
Ben tredici tasti: 0-9, punto decimale, segno negativo e l'indispensabile tasto Enter.
Key Pad non occupa slots, si installa facilmente e non richiede alcuna modifica ai programmi esistenti.
Key Pad è l'accessorio utile per: Data Entry, Calcolo, Gestione ecc...
Completo e funzionante a L. 112.000 + IVA.

NOTIZIE computer

Fatti, novità, avvenimenti, curiosità, notizie del mondo del computer

VIC-1001: il nuovo Commodore arriva dal Giappone.

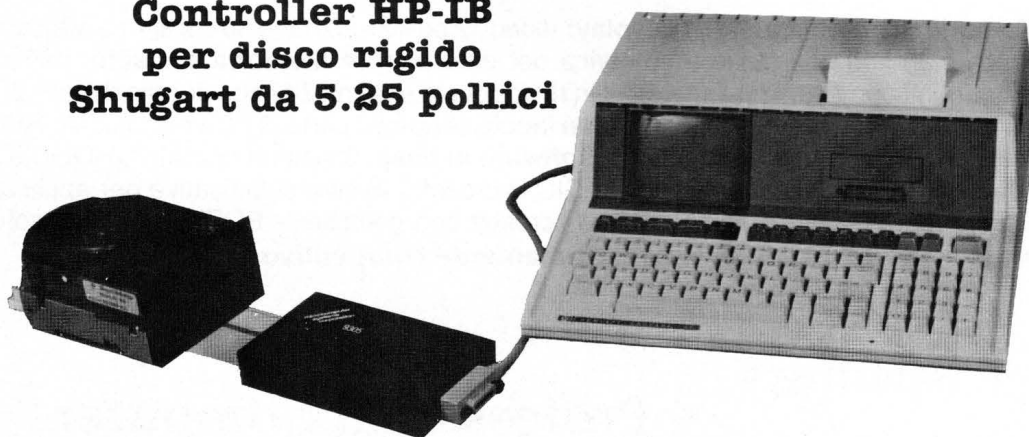
Altre novità dall'Italia.

Ed ecco il nuovo computer Commodore a basso costo: integrati nella tastiera, l'unità centrale da 4 Kbyte (espandibile a 32 K), il BASIC 4.0, l'interfaccia per registratore a cassette, l'interfaccia IEEE 488 e il modulatore video per collegarsi direttamente ai morsetti di antenna del televisore di casa. Il tutto, naturalmente, a colori. Dovrebbe essere disponibile nella primavera/estate 1981 a prezzi molto interessanti: meno di mezzo milione. Ben visibili sulla tastiera i caratteri dell'alfabeto Giapponese: il primo paese in cui è stato distribuito è per l'appunto il Giappone.

Altra novità annunciata dalla Harden, distributore Commodore per l'Italia, una scheda grafica sviluppata in Italia che connessa a qualsiasi modello della serie 2000, 3000, 4000 consente di operare in modo grafico su 320 x 200 punti. La scheda è completa della ROM con l'implementazione dei nuovi comandi grafici eseguibili sia direttamente sia sotto controllo del programma.

I comandi consentono di passare da presentazione grafica a testi e viceversa, di disegnare su fondo scuro o in negativo, di tracciare direttamente linee intere, punti, linee tratteggiate, di scrivere caratteri alfabetici sullo schermo grafico, di copiare la memoria grafica su e da disco. Non si sa ancora nulla del prezzo e della disponibilità. I primi prototipi sono però già in funzione e ci auguriamo che la scheda entri al più presto in produzione: l'interesse per la computer grafica sale di giorno in giorno.

Per informazioni: Harden - Sospiro (Cremona).

Riferimento servizio lettori: 1

Controller HP-IB per disco rigido Shugart da 5.25 pollici

La Microcomputer System Corporation di Sunnyvale (California) ha annunciato la disponibilità del controller MSC-9305 per dischi rigidi Shugart da 5 pollici e 1/4 ST-506. La cosa più interessante è che l'MSC-9305 si connette direttamente al BUS IEEE 488 (noto anche come GP-IB o HP-IB).

A titolo di esempio il costruttore ne propone l'impiego con l'HP-85 (vedi foto), con il Commodore PET e tutti i personal computer in grado di ospitare una interfaccia IEEE 488.

Il prezzo del controller MSC-9305 è di 700 dollari e le consegne sono già iniziate. Come opzioni la Microcomputer System Corporation può offrire test diagnostici per il disco, i dischi e i cavi di alimentazione.

Per informazioni: Microcomputer System Corporation - 432 Lakeside Drive - Sunnyvale, CA 94086.

Riferimento servizio lettori: 2

CONTROLLER CAMEO
e disco rigido 5+5 MByte per TRS-80
(e altri computer)

Con una lettera di intenti sottoscritta da Charles Curry della Cameo Data Systems e controfirmata da Domenico Mastromarino, la All 2000 Computer Systems si è assicurata la distribuzione in esclusiva per l'Italia del controller Cameo per sistemi a dischi rigidi fisso+mobile della Dynex. Il controller è composto di due pezzi: 1 da inserire all'interno del TRS-80 modello II (o di altri computer con microprocessore Z-80 come Cromenco, TRS-80 modello I, North Star, computer con bus S-100 etc.), l'altra destinata a funzionare in prossimità del disco. Per il momento il controller Cameo può pilotare fino a 4 dischi 5+5 Mbyte (piccola densità) per un massimo di 40 M byte, ma in futuro sarà disponibile una versione doppia densità per 4 dischi da 10+10 Mbyte con un massimo di 80 Mbyte in linea.
Come sistema operativo possono essere impiegati il TP/M, recentissima versione del CP/M che supporta un Business Basic da 18 K, l'MP/M e il CP/M.
Il prezzo dell'insieme costituito da disco rigido Dynex 5+5 Mbyte, controler Cameo e sistema operativo è stato fissato in 7.500.000 Lire all'utente finale. Una cifra particolarmente allettante.

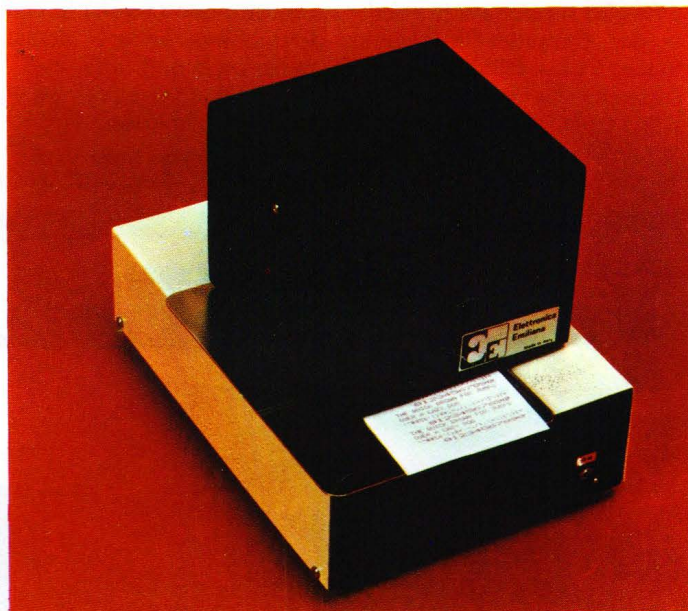
Per informazioni: All 2000 Computer Systems - Via dell'Alloro 22/A - Firenze

Riferimento servizio lettori: 3

ALFAPRINTER:
dopo il successo da nuda ci riprova vestita

Dopo il successo ottenuto con l'Alfaprinter, stampante «nuda» con meccanica giapponese Epson e interfaccia intelligente progettata in Italia (vedi m&p COMPUTER 5), l'Elettronica Emiliana S.n.c. propone ora il modello vestito della versione per ricevute fiscali e bolle di accompagnamento: si chiama Alfapi 24C e stampa le ricevute fiscali di qualsiasi formato ed i moduli discreti in genere con 35 caratteri per riga, alla velocità di 86 car/sec.
È possibile stampare, con buona qualità, fino a 5 copie ed ha di serie il sensore di presenza del modulo ed il document-stop, per scrivere con precisione su moduli prestampati.
Nessun problema per il collegamento: «alfapi» riceve i dati in ASCII parllelo; Centronics; IEEE bus 488 ed è perciò direttamente collegabile a molti personal computer tra cui Pet, Apple, Sharp.
Per controllare il buon funzionamento della stampante basta premere il pulsante di «autotest».
Elettronica Emiliana s.n.c. cura l'assistenza tecnica per l'Italia della EPSON (Seiko).

Per informazioni: Elettronica Emiliana, Via Nicoli, 53 - 41100 Modena - tel. (059) 223706.

Riferimento servizio lettori: 4

Borsa di studio per studenti universitari in chimica o chimica industriale

La Società Chimica Italiana, d'intesa con la Società Perkin-Elmer Italiana alla quale si deve l'iniziativa e il relativo finanziamento, bandisce un concorso per una borsa di studio dell'ammontare di 2 milioni di lire che dovrà essere utilizzata per svolgere un periodo di tirocinio e ricerca della durata di otto mesi presso un istituto universitario.
Al concorso possono partecipare gli studenti universitari in chimica o chimica industriale che durante l'anno accademico 1980-81 intendono preparare la tesi sperimentale di laurea con l'uso determinante di una delle seguenti tecniche strumentali: spettrofotometria IR, ivi comprese le tecniche di elaborazione dei dati forniti da tali strumenti; spettrofotometria nell'UV e visibile; spettrofotometria di fluorescenza; spettrofotometria di assorbimento atomico e di emissione con sorgente a plasma; gascromatografia (comprese le tecniche di elaborazione dei dati); cromatografia liquido-liquido ad alta risoluzione (HPLC) (incluse le tecniche di elaborazione dei dati); analisi termica (DSC-DTA-TGA); analisi organica elementare (C,H,N,O,S); spettroscopia Auger ed ESCA.
La direzione della Perkin-Elmer Italiana ha reso noto che l'iniziativa potrebbe divenire annuale.
Il bando di concorso dovrà essere richiesto alla *Società Chimica Italiana, Viale Liegi 48, 00198 Roma.*

Riferimento servizio lettori: 5

ALF: supporto software ai concessionari G.B.C. del Sud Italia.

Il 26 settembre (ma la notizia ci è giunta solo a metà novembre), è stato siglato un accordo fra la Harden, distributrice per l'Italia dei computer CBM-Commodore, e la ALF di Bari, società di software e sistemi informativi, riguardante l'assistenza ed il supporto alla commercializzazione dei suddetti computer per i concessionari G.B.C. del Sud Italia.

L'accordo nasce in visione del notevole sviluppo che questo campo di attività sta avendo su tutto il territorio nazionale.

Riportiamo integralmente un paragrafo tratto dalla comunicazione della Harden, fatta pervenire tramite la ALF a tutti i concessionari G.B.C. del Sud Italia:

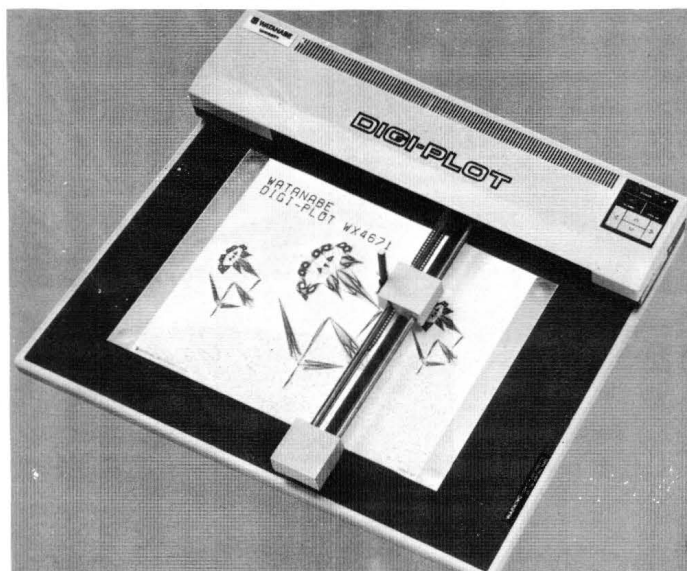
«... Al fine di razionalizzare e meglio supportare la vendita dei computer in oggetto (n.d.r. PET - CBM) che presuppone una assistenza software ed hardware efficace e la più vicina possibile, abbiamo incaricato la soc. ALF di Bari di sovrintendere la distribuzione per la zona di Vs. competenza».

La ALF si era già in precedenza occupata di supporto software per personal computer e precisamente per il Mistral 801, vale a dire della versione italiana del Pecos One la cui costruzione, dopo qualche centinaio di esemplari, è stata ormai abbandonata sull'esempio di quanto fatto in America dalla stessa APF che, dopo aver ceduto disegni e licenza di costruzione ad un operatore italiano, ha cessato la costruzione del sia pur interessante Pecos One e ha iniziato a produrre «The Imagination Machine».

Data la diffusione del PET e la necessità per i concessionari G.B.C. di un supporto software, questa volta le cose dovrebbero andare decisamente meglio.

Per informazioni: ALF S.r.l. - Via Capruzzi, 184 - 70126 Bari.

Riferimento servizio lettori: 6

MIPLOT cambia nome ora si chiama DIGI-PLOT

Lo riconoscete? Si tratta sempre del plotter Watanabe WX 4671 ampiamente descritto sul numero 6 di m&p COMPUTER, cioè del Miplot che, sembra per grane legali negli USA, ha dovuto cambiare nome: ora si chiama Digi-Plot. Tutte le altre caratteristiche (area di scrittura 360 x 260 mm, 12 comandi ad alto livello, interfaccia parallela) restano invariate.

Per informazioni: Ecta - Via Giacosa, 3 - 20127 Milano.

Riferimento servizio lettori: 7

Un centro di ricerca per l'infomatica musicale nell'area di Napoli

L'IASM, Istituto per l'Assistenza allo Sviluppo del Mezzogiorno ha presentato nel corso di un incontro svoltosi il 24 Ottobre a Napoli, una interessante proposta per la creazione di un centro di ricerca per l'informatica musicale. Il progetto denominato «operazione TECNOVA» è stato elaborato dalla TECNETRA di Milano e prevede la creazione di un centro in grado di svolgere attività di ricerca, sviluppo e assistenza tecnologica in un campo, quello dell'informatica applicata alla musica, che appare particolarmente stimolante sia sotto l'aspetto culturale che sotto quello economico.
L'Italia è uno dei maggiori produttori di strumenti musicali e sarebbe un grosso danno se sotto l'incalzare dell'industria straniera (in particolare giapponese) i nostri costruttori perdessero quote di mercato nel campo degli elettromusicali. Per scongiurare questo pericolo è necessario preparare tecnici e musicisti specializzati. Vista sotto questo aspetto l'iniziativa appare del massimo interesse. La RAI (Napoli è sede di un centro di produzione) ha offerto la propria assistenza, il progetto appare redatto con cura, il Comune di Napoli ha espresso la propria disponibilità. Non resta che augurarsi che il tutto abbia un seguito.

Per informazioni: Istituto per l'Assistenza allo Sviluppo del Mezzogiorno, Viale Pilsudski, 124 - 00197 Roma.

Riferimento servizio lettori: 8

Tornei di Bridge: un personal per le classifiche

In un recente torneo di bridge a coppie svoltosi presso il CIRCOLO NERAZZURRO di Milano, è stato utilizzato un «PET» nelle mansioni di giudice di classifica.
Senza l'ausilio del computer, i risultati di questi tornei vengono generalmente a conoscenza dei giocatori dopo alcuni giorni dalla loro effettuazione; con il computer ed il programma preparato dalla LOTUS, i medesimi risultati si possono avere dopo qualche decina di minuti (il tempo di prendere un caffè al bar) con comprensibile grande interesse ed entusiasmo dei partecipanti al torneo. (Oggi si parla con sempre maggiore insistenza di «tempo reale»).
La configurazione minima richiesta è costituita da un computer da 32 K, registratore a cassette e stampante.
Nella presentazione è stato utilizzato un PET 3032, con relativa stampante e cassetta.

Per informazioni: LOTUS Via Padova, 217 - Milano.

Riferimento servizio lettori: 9

STAMPANTI GENERAL ELECTRIC DISTRIBUITE DA MACTRONIC

La Mactronic Italia ha stipulato un accordo con la General Electric per la distribuzione in Italia di stampanti e terminali scriventi (un terminale scrivente è una stampante dotata di tastiera e di una interfaccia seriale particolarmente completa). I terminali impiegano 2 microprocessori 8085 ed hanno una velocità compresa tra 30 e 120 caratteri al secondo. La gamma delle stampanti comprende modelli a matrice da 200 c/s e a cinghia da 340 righe al minuto.
Un'altra novità degna del massimo rilievo è una stampante per word processing con qualità di stampa paragonabile a quella delle stampanti a margherita ma di velocità elevatissima: 510 caratteri/secondo, oltre 100 volte quella di una stampante a margherita.

Per informazioni: Mactronics Italia - Viale Jenner, 40 A - 20159 Milano

Riferimento servizio lettori: 10

PISA 1-2 GIUGNO 1981 Quarto colloquio di informatica musicale

Da qualche anno un gruppo sempre crescente di persone interessate allo sviluppo delle ricerche nel campo interdisciplinare fra *musica* ed *informatica* (musicisti, ricercatori, docenti, studenti...), operanti prevalentemente presso Conservatori, Università, Istituti del Consiglio Nazionale delle Ricerche, ed altri enti pubblici e privati, sente l'esigenza di riunirsi periodicamente per un aggiornamento reciproco sulle ricerche in corso, per uno scambio di esperienze e, contemporaneamente, per promuovere ed allargare il dibattito sul tema «musica ed elaboratore elettronico».
Durante questi incontri, denominati Colloqui di Informatica Musicale (Pisa, CNUCE-C.N.R., 23-24/2/76; Milano, Istituto di Cibernetica, 14/12/77; Padova, Università, 2-3/4/79), aperti a tutti gli interessati, sono state presèntate e discusse memorie di carattere sia teorico che applicativo, riguardanti principalmente i seguenti temi:
— analisi musicologica mediante computer
— tecniche di analisi, di elaborazione e di sintesi del suono
— programmi e linguaggi di programmazione per la generazione, la codifica, l'elaborazione e l'esecuzione di strutture musicali
— rappresentazione formale di strutture musicali
— simulazione di processi compositivi
— esperienze di musicisti e docenti sull'impiego di vari sistemi di computer music esistenti in Italia.

Per ricevere la scheda di adesione scrivere a: Quarto colloquio di informatica musicale - Reparto di musicologia, CNUCE-CNR, Via S. Maria, 36 - 56100 Pisa.

Riferimento servizio lettori: 11

libricomputer

M. MOROCUTTI
R. CERRUTTI,
INTRODUZIONE AI MICROPROCESSORI
Hoepli, Milano, 1980,
pp. 108, L. 6.000.

Casa Editrice:
Ulrico Hoepli Editore
Via Hoepli, 5 - 20121 Milano

Riferimento servizio lettori: 30

Non è necessaria una grossa cultura informatica per sapere che i microprocessori rappresentano il cuore di ogni microcomputer. È quindi evidente la loro importanza al fine di una conoscenza un po' approfondita del sistema microcomputer: d'altra parte, dal momento che la tecnologia consente la produzione di dispositivi sempre più piccoli, potenti, a costi decrescenti e con un alto grado di affidabilità, è facile comprendere la diffusione sempre più massiccia dei microprocessori in applicazioni tra le più diverse, giustificando così l'attenzione di cui questi minuscoli dispositivi sono fatti oggetto.
La tecnologia costruttiva, basata su tecniche LSI (Large Scale Integration), è impostata su procedimenti ormai consolidati, assai vicini ai limiti fisici dei materiali usati, e, a meno di un radicale cambiamento tecnologico, non offre molti spazi innovativi: assai diverso è invece il problema delle applicazioni, in termini di flessibilità e potenza dei sistemi, ed è questo anche il settore che ci interessa più da vicino. I microprocessori in commercio sono numerosi, ma in quanto a caratteristiche sono raggruppabili in varie «famiglie», mentre la tipologia di funzionamento è molto simile per tutti. Proprio questi due aspetti (funzionamento e caratteristiche) sono affrontati direttamente dal libro che presentiamo: certo il livello cui sono tenute le informazioni non è particolarmente elevato, come d'altronde imposto dalla volontà espressa dagli autori di non fare un testo per specialisti, ma tutto è sempre chiaro e completo (riferendo tale termine, ovviamente, ai limiti di «progetto» del testo).
La prima parte, centrata nel «modus operandi» del microprocessore, illustra molte funzioni e peculiarità dei dispositivi, ricorrendo a molte instruzioni in assembler, sia per quanto concerne le unità di elaborazione che per quelle di memoria e input-output. La seconda parte, più «pratica», è occupata dalla descrizione dettagliata di alcuni tra i più diffusi microprocessori del mercato, nonché da un'appendice con alcuni semplici programmi illustrativi per MCS 6502, scritti in assembler e adeguatamente commentati.
Globalmente il testo risulta dunque assai interessante, soprattutto per chi voglia comprendere con un certo dettaglio le strutture dell'elaborazione, pur non disponendo di conoscenze specifiche.

Gualtiero Rudella

MADE IN ITALY

personaggi & interpreti

La facilità d'uso del modello T apre le porte dell'informatica anche ai non esperti.
Chiunque può usarlo e soprattutto programmarlo in rapporto alla propria attività, piccola o grande che sia. I vantaggi sono presto valutabili: massima adattabilità, costi di gestione quasi inesistenti, facilità di manutenzione, ingombro contenuto.
La General Processor è la prima azienda italiana produttrice di elaboratori personali che per la loro moderna concezione, per la loro massima affidabilità ed il costo decisamente competitivo, rappresentano quanto di meglio e di nuovo offra oggi il mercato.
Il modello T è stato progettato per adattarsi alle esigenze dell'utente; la sua flessibilità e la sua modularità rendono possibile la scelta della configurazione più adatta alle condizioni operative. Quattro modelli diversi ne permettono l'uso sia al professionista (ingegnere, ricercatore scientifico, ecc.) sia alla piccola e grande azienda.
Il modello T è compatibile col noto sistema operativo CP/M[tm]; da ciò consegue la possibilità di un accesso immediato ad una delle più estese biblioteche di programmi a livello mondiale. Con un apposito programma si ha la possibilità di convertire i dati per la perfetta compatibilità con i sistemi IBM.

MODELLO "T"

SE DESIDERA MAGGIORI INFORMAZIONI SUL MODELLO T SCRIVA ALLA GENERAL PROCESSOR ALLEGANDO QUESTO VIDEO-COUPON

MP

GP

GENERAL PROCESSOR

pensato, progettato, costruito in Italia

GENERAL PROCESSOR s.r.l. / SISTEMI DI ELABORAZIONE / VIA PIAN DEI CARPINI, 1 / TEL. 055-435527 / 50127 - FIRENZE

FIRENZE
ALL 2000 COMPUTER SYSTEMS
055/283772-268396 -Telex 572507

MILANO
3 R ELECTRONICS MANAGEMENT
02/793471

P G E SISTEMI INFORMATIVI
02/2822225

BRESCIA
SIBIESSE
030/661006

BERGAMO
MICROTEM
035/241862

TREVISO
S.H.A.
0438/87301

TRIESTE
Ditta MURRI
040/65630

CARPI (MO)
Ditta MESCHIARI
059/683574

FORLÌ
TECNO UFFICIO
0543/35855

CESENA (FO)
ST. AUT. DI GUIDUCCI & C.
0547/24800

GENOVA
ELAB 80
010/200738

PISTOIA
CEIA SYSTEMS
0572/51611

PRATO (FI)
GERVA SYSTEMS
0574/592694

S. CROCE SULL'ARNO (PI)
ELETTROTECNICA DAINELLI
0571/31805

LIVORNO
CED 05
0586/25395

ROMA
DITTA S.I.S.M.
06/351377

FORMIA (LT)
CONTAX s.r.l.
0771/22503-26302

NAPOLI
COMPU SYSTEMS s.r.l.
081/463602

TECNODATA
081/242166

SHADO
081/7267412

Interprete GD 010 LISP
IL LISP A PORTATA DI PET

Nato inizialmente come un linguaggio orientato soltanto alla gestione delle liste, il LISP è invece risultato il primo passo verso un calcolatore dotato di logica deduttiva, meritando l'appellativo di «intelligenza artificiale». Questo potente ed interessante strumento, destinato ad avere sempre maggior rilievo, è ora alla portata di tutti: la Commodore distribuisce un interprete LISP, su dischetto, per il 3032.

Distributore:
Harden S.p.A.
26048 Sospiro (Cremona)
Prezzo: L. 250.000
Riferimento servizio lettori: 31

Avevo scritto su queste pagine, a conclusione dell'ultimo articolo sull'intelligenza artificiale, come sarebbe stato di grande aiuto per lo sviluppo dell'informatica sui piccoli calcolatori che un linguaggio come il LISP — semplice, schematico, e soprattutto lontano dalle implicazioni di calcolo numerico che tutti gli altri linguaggi richiedono — fosse reso disponibile su qualcuno dei personal in commercio; ma pensavo fossero pure utopie, al punto che stavo accarezzando l'idea di scrivere personalmente un programma traduttore...
Invece un giorno arriva in redazione un dischetto contenente un interprete LISP per il Pet 3032: un'occasione da non perdere, per un appassionato come il sottoscritto, e difatti dopo qualche ora... ero già al lavoro con un esauriente manuale di istruzioni.
Mi sono divertito un bel po' e, quando ho iniziato a tirare le somme delle prove effettuate, mi sono accorto che non solo avevo elementi sufficienti per esprimere un giudizio sull'interprete, ma che avevo anche elaborato alcuni risultati abbastanza utili per comprendere la filosofia ed alcune prospettive e applicazioni della programmazione in LISP.
Questo articolo, riassunto delle prove che ho svolto sul Pet, ha dunque il duplice scopo di illustrare le caratteristiche dell'interprete GD 010 LISP, e di procedere sul cammino, già intrapreso nei precedenti articoli, volto a studiare alcune interessanti applicazioni dell'intelligenza artificiale nel campo di impiego dei piccoli calcolatori.

Il sistema: un'occhiata al linguaggio

Le funzioni offerte dal GD 010 LISP sono in numero maggiore e più estese di quelle del LISP originale, e soprattutto di quelle descritte nel mio primo articolo sull'intelligenza artificiale: conviene quindi prima di tutto elencare le principali estensioni dell'interprete rispetto al linguaggio «classico», in modo da comprendere appieno le grosse facilitazioni offerte anche a chi non é pratico di programmazione strutturata e ricorsiva.
Nel seguito si assumerà comunque noto il LISP «classico», ed in particolare la struttura e il funzionamento delle sue istruzioni fondamentali (QUOTE, COND, LAMBDA).
Le funzioni aggiunte dal GD 010 LISP si possono dividere in tre categorie:
— funzioni che riassumono sequenze di operazioni ripetitive, o comunque scomode per l'utente, snellendo il linguaggio;
— funzioni rivolte alla gestione del sistema e delle sue periferiche;
— funzioni estensive, che rendono il linguaggio adoperabile anche da chi ha dimestichezza con metodi di programmazione più classici.
Al primo gruppo appartengono un gran numero di funzioni, che rendono il linguaggio molto ricco e versatile.
Innanzitutto é facilitata la *definizione* delle funzioni, che nel LISP classico si esegue con l'espressione LAMBDA. La funzione (DE (parametri)(espressione)) definisce una LAMBDA — espressione come funzione dei parameri elencati. Dove nel LISP classico occorreva scrivere:
(DEF funzione (LAMBDA (parametri)(espressione)))
nel GD 010 basta scrivere:
(DE funzione (parametri)(espressione)),
e l'interprete pensa da solo ad inserire l'atomo LAMBDA, che difatti compare nei listati dei programmi presenti in questo articolo.
I parametri possono essere in forma di *atomi* o di *coppie puntate*. Se si trovano in quest'ultima forma, il CDR della coppia esprime il valore di «default», che la funzione assume automaticamente nel caso che il parametro non sia pre-

sente nella lista di chiamata.
Se per esempio si definisce una funzione in questo modo:
(DE FUNZ (X (Y.1))(...))
una chiamata (FUNZ 1 2) genera una elaborazione della funzione con i parametri X=1 e Y=2, mentre una chiamata (FUNZ 5) genera una elaborazione con X=5 e Y=1 (valore di «default»).
Il GD 010 LISP permette inoltre di definire delle variabili locali tramite le funzioni SET e SETQ. Il loro funzionamento é uguale a quello del segno «=» nel BASIC, ed esse assegnano un valore ad un atomo. La differenza fra SET e SETQ sta nel fatto che la prima valuta *anche il nome della variabile,* mentre la seconda prende per buono ciò che trova nell'espressione.
Ad esempio, si voglia definire un numero complesso nelle sue parti reale ed immaginaria. Il modo più semplice consiste nel costruire una coppia puntata COMPLEX formata dai due atomi REAL e IMAG:
(SETQ COMPLEX (CONS REAL IMAG)).
In questo caso abbiamo usato SETQ, poiché «COMPLEX» é effettivamente il nome della variabile che vogliamo definire.
Supponiamo ora di volere assegnare a COMPLEX il valore 1+i2, ossia 1 alla parte reale e 2 a quella immaginaria:
(SET (CAR COMPLEX) 1)
(SET (CDR COMPLEX) 2).
In questo caso abbiamo usato SET, poiché il nome della variabile (risp. REAL e IMAG) é ricavato come *valore* di un'espressione LISP (risp. CAR e CDR della coppia puntata COMPLEX).
Un'altra grossa facilitazione é data dall'interprete con la coppia di funzioni EXPLODE e IMPLODE: la prima di esse trasforma un atomo in una lista formata dai caratteri che lo compongono, e viceversa la seconda trasforma una lista di caratteri in un atomo.
«Esplodendo» *l'atomo* 'PIETRO si ottiene la *lista (P I E T R O), e «implodendo» questa lista si riottiene l'atomo di partenza.*
In questo modo si possono trattare le stringhe di caratteri con le stesse procedure delle liste; ad esempio l'espressione:
(CAR (EXPLODE X))
dà come valore il primo carattere dell'atomo X, e corrisponde alla funzione BASIC LEFT (X$,1).
Per rimanere nel campo delle analogie con il BASIC, la funzione CHARACTER, con argomento numerico, dà come valore il carattere ASCII con codice uguale all'argomeno:
(CHARACTER 65) ha valore A;
mentre la funzione ORDINAL con argomento di tipo stringa dà come valore il codice ASCII del primo carattere della stringa:
(ORDINAL 'ABCDE) ha valore 65, poiché 65 é la codifica ASCII del carattere A.
Nel campo matematico é possibile eseguire moltiplicazioni con la funzione TIMES e divisioni intere con la funzione QUOTIENT, mentre la funzione REMAINDER dà come valore il *resto* della divisione intera dei suoi argomeni.
Altre due funzioni, ADD1 e SUB1, incrementano e decrementano di 1 il loro argomento, evitando una PLUS o DIFFERENCE a due argomenti.

Esiste poi una funzione LIST che dà come valore una lista formata dai suoi argomenti, evitando di dover passare attraverso una serie di CONS nidificate. Infatti, se si vuole costruire una lista formata dagli elementi A, B e C, la soluzione in LISP classico è:
(CONS A (CONS B (CONS C NIL))),
mentre il medesimo risultato é raggiunto dal GD 010 in questo modo:
(LIST A B C).
Un'altra funzione atta a snellire i calcoli é la MAP, che applica una funzione (presente in argomento) ad una lista (o coppia di liste) nel seguente modo:
— il valore dell'espressione (MAP funzione L1 [L2]) é una lista il cui primo elemento é il valore della funzione applicata ai primi elementi di L1 e L2, il secondo é il valore della funzione applicata ai secondi elementi di L1 ed L2, e così via.
Il valore di (MAP TIMES (1 2 3)(4 5 6)) é la lista (4 10 18), mentre il valore di (MAP ADD1 (1 2 3)) é la lista (2 3 4).
Il secondo gruppo di funzioni aggiunte é formato da una serie di direttive che permettono al programma di interagire con il Pet e le sue periferiche, o comunque di gestirsi la prestazione del software.
Esistono quindi le funzioni OPEN e CLOSE con lo stesso formato e significato dei corrispondenti comandi usati nel BASIC; inoltre un programma può essere caricato in memoria con una LOAD e salvato su periferica con una SAVE. Una funzione PAGELINES specifica la lunghezza in righe di una pagina sul video o sulla stampante, mentre una funzione LINEWIDTH specifica la lunghezza in caratteri della riga.
READ e PRINT sono le istruzioni di lettura e scrittura sul video, mentre PRINØ# é l'istruzione di scrittura sul file logico. Esistono anche le funzioni PEEK e POKE che leggono e scrivono direttamente una parola di memoria, e c'é anche una funzione CALL che permette di richiamare un sottoprogramma in linguaggio macchina.
Questa parte dell'interprete é realizzata con molta cura, e permette al LISP quasi tutte le istruzioni di ingresso, uscita e gestione del sistema tipiche del BASIC.
L'ultima categoria, comprendente funzioni come LOOP, WHILE e UNTIL, «inquina» in un certo senso il linguaggio, in quanto offre soluzioni alternative alla gestione ricorsiva tipica del LISP, ed é soprattutto utile a chi non é ancora esperto. Infatti queste funzioni dovrebbero, nelle intenzioni dei costruttori, semplificare la vita a chi é abituato a programmare in BASIC e PASCAL e alla loro struttura a cicli (FOR... NEXT oppure *repeat... until*). Personalmente, a rischio di venire scambiato per un purista arrabbiato, preferisco dare al PASCAL ciò che é del PASCAL ed affrontare un linguaggio originale come il LISP con la giusta filosofia, ossia scrivendo programmi *ricorsivi* piuttosto che programmi *ciclici,* e cercando il più possibile di concentrare le funzioni in una sola operazione. L'esempio di fig. 1) presenta la stessa funzione EXTR, che estrae da una lista tutti gli elementi in cui é presente un certo valore X, scritta in modo ciclico ed in modo

ricorsivo: la prima stesura é forse più chiara, ma spreca una variabile di appoggio, distrugge la lista di partenza, e soprattutto ...... non é LISP!!!

In conclusione, il linguaggio offre una serie di prestazioni che vanno ben al di là del LISP classico, e che permettono una programmazione semplice, schematica ed efficace nei più disparati campi dell'informatica, da quello logico-matematico, alla gestione delle banche di dati, agli esperimenti di intelligenza artificiale. Se un difetto si può trovare, risiede nella scarsa protezione (e fa meraviglia in un sistema tanto sofisticato) da errori nei trasferimenti o dall'uso sbagliato delle parole riservate. Mi é capitato ad esempio di definire una variabile LIST, e di perdermi così completamente la *funzione* LIST, senza che l'interprete se ne accorgesse: salvo segnalarmi a cose fatte un errore nel programma, poiché LIST significava ormai per lui un certo insieme di dati definito da me, e non più una funzione di sistema.

I programmi: qualche applicazione

E passiamo finalmente ad esaminare in pratica le possibilità di questo sorprendente e versatile linguaggio: grazie all'universalità della struttura a lista é possibile spaziare attraverso campi di interesse diversissimi e risolvere problemi anche abbastanza gravosi in modo semplice e immediato. Basta seguire un'avvertenza: cercare di entrare nella logica del LISP, e, come ho detto più sopra, di costruire programmi ricorsivi ed estendibili; anche se a prima vista può sembrare più complesso, con un pò di pratica un simile approccio ai problemi risulta conveniente ed efficace.

Verranno presentati di seguito tre gruppi di funzioni che affrontano tre tipi di problematiche completamente diverse con la stessa logica LISP, e precisamente la creazione della derivata di una funzione matematica, la gestione di una semplice banca di dati, ed alcune operazioni sugli insiemi con una applicazione alla teoria dei grafi.

Un excursus matematico: la derivata F'(x)

Chi ha studiato analisi matematica ricorderà quanto é penoso, nello studio delle funzioni, il calcolo delle derivate prima e seconda: c'é il costante rischio di perdersi per strada, specie quando la funzione é composta. E il guaio é che l'avvento dell'informatica non ha portato alcun valido aiuto a questo tipo di calcolo, poiché la macchina possiede eccellenti algoritmi per calcolare la derivata di una funzione in un *punto*, e darne il *il valore numerico*, ma non é mai stata capace di eseguire calcoli in *funzione* di una variabile x.

Il problema é invece di facile soluzione in LISP, dove fra istruzioni e dati non c'é differenza, e una funzione può quindi essere vista anche come una lista da manipolare a piacere. Ed è appunto ciò che fa la funzione DERIV descritta nel progamma 1.

Questa funzione opera su un unico argomento X, che é una funzione matematica della variabile x. Data la ricorsività del linguaggio, questa funzione sarà sempre nella forma:

```
↑VALUE IS : 5

↑EVALUATE : (LAMBDA
   (X L)
   (SETQ Z NIL)
   (LOOP
      (UNTIL (NULL L) Z)
      (COND
         ((PRES X (CAR L))
            (SETQ Z (CONS Z (CAR L))))
         (T 1))
      (SETQ L (CDR L))))
↑VALUE IS : )


↑EVALUATE :
↑VALUE IS : 5

↑EVALUATE : (LAMBDA
   (X L)
   (COND
      ((NULL L) NIL)
      ((PRES X (CAR L))
         (CONS
            (CAR L)
            (EXTR X (CDR L))))
      (T (EXTR X (CDR L)))))
↑VALUE IS : )
```

Fig. 1 - Una funzione LISP scritta in forma ciclica e in forma ricorsiva: la seconda scrittura, anche se a prima vista più complessa, è più compatta ed efficiente.

```
↑EVALUATE :
↑VALUE IS : DERIV

↑EVALUATE : (LAMBDA
   (X)
   (COND
      ((ATOM X) (DERIV (CONS X NIL)))
      ((OR
         (EQ (CAR X) (QUOTE PLUS))
         (EQ
            (CAR X)
            (QUOTE DIFFERENCE)))
       (LIST
         (CAR X)
         (DERIV (CADR X))
         (DERIV (CADDR X))))
      ((EQ (CAR X) (QUOTE TIMES))
       (LIST
         (QUOTE PLUS)
         (LIST
            (QUOTE TIMES)
            (DERIV (CADR X))
            (CADDR X))
         (LIST
            (QUOTE TIMES)
            (DERIV (CADDR X))
            (CADR X))))
      ((EQ (CAR X) (QUOTE QUOTIENT))
       (LIST
         (QUOTE QUOTIENT)
         (LIST
            (QUOTE DIFFERENCE)
            (LIST
               (QUOTE TIMES)
               (DERIV (CADR X))
               (CADDR X))
            (LIST
               (QUOTE TIMES)
               (DERIV (CADDR X))
               (CADR X)))
         (LIST
            (QUOTE ELEV)
            (CADDR X)
            2)))
      ((EQ (CAR X) (QUOTE LOG))
       (LIST
         (QUOTE QUOTIENT)
         (DERIV (CADR X))
         (CADR X)))
      ((EQ (CAR X) (QUOTE ELEV))
       (LIST
         (QUOTE TIMES)
         (LIST
            (QUOTE TIMES)
            (CADDR X)
            (LIST
               (QUOTE ELEV)
               (CADR X)
               (SUB1 (CADDR X))))
         (DERIV (CADR X))))
      ((EQ (CAR X) (QUOTE X)) 1)
      (T 0)))
↑VALUE IS : )
↑EVALUATE :
↑VALUE IS : (PLUS X 2)

↑EVALUATE :
↑VALUE IS : (PLUS 1 0)


↑EVALUATE :
↑VALUE IS : (ELEV (PLUS (TIMES X 2) 1) 2)

↑EVALUATE :
↑VALUE IS : (TIMES (TIMES 2 (ELEV
(PLUS (TIMES X 2) 1) 1)) (PLUS
(PLUS (TIMES 2) (TIMES 0 X)) 0))

↑EVALUATE : (TIMES
   (TIMES
      2
      (ELEV
         (PLUS
            (TIMES X 2)
            1)
         1))
   (PLUS
      (PLUS
         (TIMES 1 2)
         (TIMES 0 X))
      0))
↑VALUE IS : )
```

Programma n. 1: la funzione DERIV calcola la derivata di una funzione matematica X.

```
↑EVALUATE :
↑VALUE IS : 5

↑EVALUATE :
↑VALUE IS : PRES

↑EVALUATE : (LAMBDA
   (X L)
   (COND
      ((NULL L) NIL)
      ((ATOM L) (EQ X L))
      ((EQ X (CAR L)) T)
      (T (PRES X (CDR L)))))
↑VALUE IS : )

↑EVALUATE :
↑VALUE IS : ABS

↑EVALUATE : (LAMBDA (X L) (NULL (PRES X L)))
↑VALUE IS : (LAMBDA (X L) (NULL (PRES X L)))

↑EVALUATE :
↑VALUE IS : UNION

↑EVALUATE : (LAMBDA
   (A B)
   (COND
      ((NULL B) A)
      ((ATOM A)
         (UNION B (CONS A NIL)))
      ((ATOM B)
         (UNION A (CONS B NIL)))
      ((ABS (CAR B) A)
         (CONS
            (CAR B)
            (UNION A (CDR B))))
      (T (UNION A (CDR B)))))
↑VALUE IS : )


↑EVALUATE :
↑VALUE IS : INTERS

↑EVALUATE : (LAMBDA
   (A B)
   (COND
      ((NULL A) NIL)
      ((PRES (CAR A) B)
         (CONS
            (CAR A)
            (INTERS (CDR A) B)))
      (T (INTERS (CDR A) B))))
↑VALUE IS : )


↑EVALUATE :
↑VALUE IS : DIFF

↑EVALUATE : (LAMBDA
   (A B)
   (COND
      ((NULL A) NIL)
      ((ABS (CAR A) B)
         (CONS
            (CAR A)
            (DIFF (CDR A) B)))
      (T (DIFF (CDR A) B))))
↑VALUE IS : )
```

Programma n. 3: funzioni svolgenti le principali operazioni sugli insiemi.

```
↑EVALUATE :
↑VALUE IS : 5

↑EVALUATE :
↑VALUE IS : PREC

↑EVALUATE : (LAMBDA
   (Y X)
   (COND
      ((NULL X) NIL)
      ((NULL Y) T)
      ((EQ (ORDINAL X) (ORDINAL Y))
         (PREC (RESTO Y) (RESTO X)))
      (T
         (GREATERP
            (ORDINAL X)
            (ORDINAL Y)))))
↑VALUE IS : )

↑EVALUATE :
↑VALUE IS : RESTO

↑EVALUATE : (LAMBDA
   (Y)
   (COND
      ((NULL (EXPLODE Y)) NIL)
      (T (IMPLODE (CDR (EXPLODE Y))))))
↑VALUE IS : )

↑EVALUATE :
↑VALUE IS : INS

↑EVALUATE : (LAMBDA
   (X L)
   (COND
      ((NULL L) (CONS X NIL))
      ((PREC (CAR X) (CAAR L))
         (CONS X L))
      (T
         (CONS
            (CAR L)
            (INS X (CDR L))))))
↑VALUE IS : )

↑EVALUATE :
↑VALUE IS : ADD

↑EVALUATE : (LAMBDA
   (X)
   (SETQ LISTA (INS X LISTA)))
↑VALUE IS : )

↑EVALUATE :
↑VALUE IS : TROVA1

↑EVALUATE : (LAMBDA
   (X L)
   (COND
      ((NULL L) NIL)
      ((MATCH X (CAR L))
         (CONS
            (CAR L)
            (TROVA1 X (CDR L))))
      (T (TROVA1 X (CDR L)))))
↑VALUE IS : )

↑EVALUATE :
↑VALUE IS : TROVA

↑EVALUATE : (LAMBDA (X) (TROVA1 X LISTA))
↑VALUE IS : (LAMBDA (X) (TROVA1 X LISTA))

↑EVALUATE :
↑VALUE IS : DEL1

↑EVALUATE : (LAMBDA
   (X L)
   (COND
      ((NULL L) NIL)
      ((MATCH X (CAR L))
         (DEL1 X (CDR L)))
      (T
         (CONS
            (CAR L)
            (DEL1 X (CDR L))))))
↑VALUE IS : )

↑EVALUATE :
↑VALUE IS : DEL

↑EVALUATE : (LAMBDA (X) (DEL1 X LISTA))
↑VALUE IS : (LAMBDA (X) (DEL1 X LISTA))

↑EVALUATE :
↑VALUE IS : ESEMPIO

↑EVALUATE :
↑VALUE IS : (SETQ LISTA NIL)

↑EVALUATE :
↑VALUE IS : NIL

↑EVALUATE :
↑VALUE IS : (ADD (QUOTE (PIETRO MILANO 1953)))

↑EVALUATE :
↑VALUE IS : ((PIETRO MILANO 1953))

↑EVALUATE :
↑VALUE IS : (ADD (QUOTE (ALBA ROMA 1954)))

↑EVALUATE : ((ALBA ROMA 1954)
   (PIETRO MILANO 1953))
↑VALUE IS : )

↑EVALUATE :
↑VALUE IS : (ADD (QUOTE (GIULIA ROMA 1930)))

↑EVALUATE : ((ALBA ROMA 1954)
   (GIULIA ROMA 1930)
   (PIETRO MILANO 1953))
↑VALUE IS : )

↑EVALUATE :
↑VALUE IS : (ADD (QUOTE (GIULIANA MILANO 1921)))

↑EVALUATE : ((ALBA ROMA 1954)
   (GIULIA ROMA 1930)
   (GIULIANA MILANO 1921)
   (PIETRO MILANO 1953))
↑VALUE IS : )

↑EVALUATE :

↑VALUE IS : (TROVA (QUOTE (? MILANO ?)))

↑EVALUATE : ((GIULIANA MILANO 1921)
   (PIETRO MILANO 1953))
↑VALUE IS : )

↑EVALUATE :
↑VALUE IS : (ADD (QUOTE (NUCCIO ROMA 1930)))

↑EVALUATE :
↑VALUE IS : (ADD (QUOTE (PAOLO MILANO 1908)))

↑EVALUATE :
↑VALUE IS : 5

↑EVALUATE : ((ALBA ROMA 1954)
   (GIULIA ROMA 1930)
   (GIULIANA MILANO 1921)
   (NUCCIO ROMA 1930)
   (PAOLO MILANO 1908)
   (PIETRO MILANO 1953))
↑VALUE IS : )

↑EVALUATE :
↑VALUE IS : (TROVA (QUOTE (? ? 1930)))

↑EVALUATE : ((GIULIA ROMA 1930)
   (NUCCIO ROMA 1930))
↑VALUE IS : )

↑EVALUATE :
↑VALUE IS : (DEL (QUOTE (? ROMA ?)))

↑EVALUATE : ((GIULIANA MILANO 1921)
   (PAOLO MILANO 1908)
   (PIETRO MILANO 1953))
↑VALUE IS : )
```

Programma n. 2: gestione di un piccolo data-base. Possono essere eseguite aggiunte, ricerche e cancellazioni.

(FUNZ (ARG 1) [(ARG2)]) (1)
gli argomenti ARG1 e ARG2 potranno a loro volta essere funzioni del tipo definito dalla (1), oppure consistere del solo parametro x, oppure essere costanti.
Sfruttando questa scrittura rigida e nidificata delle funzioni matematiche, e la ricorsività del linguaggio, la funzione DERIV costruisce la derivata della funzione X come una lista seguendo nell'ordine queste regole di derivazione:

— se la funzione è una somma F1±F2:
D(X) = D(F1) ± D(F2)
— se é una moltiplicazione F1·F2:
D(X) = D(F1)·F2 + D(F2)·F1
— se è una divisione F1/F2:

$$D(X) = \frac{D(F1)\cdot F2 - D(F2)\cdot F1}{(F2)^2}$$

— se è un logaritmo naturale $\log_e$ (F):
D(X) = D(F)/F
— se è un elevamento a potenza $(F)^n$:
$D(X) = n\ (F)^{n-1}\cdot D(F)$
— se è la variabile x:
D(X) = 1
— se é una costante:
D(X) = 0
La ricorsività del LISP fa sì che il problema venga risolto automaticamente elencando questa casistica (a cui si potrebbero aggiungere altre funzioni quali seni e coseni). Il risultato é comunque sempre una lista in forma di corret-

ta funzione matematica della variabile x, che può quindi essere posta in una LAMBDA — espressione e valutata direttamente, qualora servano calcoli numerici.
Il primo dei due esempi presentati deriva la semplice espressione (X+2) ottenendo (1+0) = 1 mentre il secondo deriva la funzione composta $(2X+1)^2$, ottenendo la funzione $2\cdot((2X+1)^1\cdot((X\cdot0+2)+0)) = 2\cdot(2X+1)\cdot2$.
Certo col crescere della complessità della funzione da derivare cresce anche il numero di operazioni ridondanti (moltiplicazioni per zero e per uno, somme con lo zero etc.); un'altra limitazione risiede nel fatto che il programma non accetta funzioni con più di due argomenti. A quest'ultima si può rimediare spezzando le lunghe somme e moltiplicazioni, nel senso che (PLUS A B C D) dà lo stesso valore di (PLUS A(PLUS B(PLUS C D))), ma in questa ultima scrittura ogni PLUS ha solo due argomenti. Il problema della riduzione della ridondanza é invece di carattere software: occorre scrivere una funzione — analoga a DERIV — che riconosca le configurazioni ridondanti e le riduca. Lascio il problema come esercizio.

Un'applicazione gestionale: chi cerca ... TROVA

Il secondo gruppo di routines, listate nel programma 2, é stato presentato al recente SMAU, e svolge le funzioni di aggiornamento e ricerca in una piccola banca di dati, riprendendo in parte un esempio già esposto in sede teorica nei precedenti articoli.
La banca di dati é formata da un'unica lista (chiamata appunto LISTA), in cui vengono inseriti degli elementi, anch'essi sotto forma di liste, contenenti ciascuno un'unità di dati. Al di sotto di questa struttura il formato della banca di dati é libero, ed in particolare é libera la struttura del singolo elemento, ed é anzi possibile avere più di un tipo di elementi nella stessa banca di dati.
Nello scrivere il programma ho pensato ad una applicazione (qualsiasi) rivolta a gestire un insieme di *persone*, per cui un singolo elemento sarà caratterizzato dal *nome* di una persona, che per comodità viene posto *all'inizio* della lista formante l'elemento stesso: tutto ciò spiega perché la banca di dati risulta organizzata secondo *l'ordine alfabetico dei CAR degli elementi.*
La prima funzione, PREC, ha quindi il compito di determinare se un atomo precede o no un altro nell'ordine alfabetico : per stabilire ciò, confronta il primo carattere dei due atomi, e se trova diversità l'algoritmo é concluso; altrimenti prosegue pari pari l'analisi sul RESTO dei due atomi (questa seconda funzione ha il solo compito di scartare il primo carattere del suo argomento).
Se uno dei due atomi termina prima di trovare una discrepanza nei caratteri (é il caso, ad esempio, dei nomi GIULIA e GIULIANA), esso viene assunto come «precedente» l'altro.
Possono ora essere inseriti nella banca di dati i nuovi elementi, e ciò si ottiene con le funzioni INS e ADD.
La prima opera su una lista parametrica (per permettere di scomporla), e la seconda richiama la precedente passandole come parametro il nome della banca di dati.
Si supponga di voler costruire un data-base con dei semplici elementi indicanti, per ogni persona, il nome, il luogo e l'anno di nascita. Basterà inizializzare a NIL la variabile LISTA:
(SETQ LISTA NIL)
ed insieme uno per volta gli elementi desiderati, ad esempio:
(ADD '(PIETRO MILANO 1953))
(ADD '(ALBA ROMA 1954))
e così via: la banca di dati risulterà organizzata secondo l'ordine alfabetico dei nomi.
Costruito così il data-base, si può pensare di effettuare delle ricerche su di esso, ad esempio trovare tutte le persone nate in un determinato luogo o in un determinato anno, e così via (é chiaro che le possibiltà si moltiplicano al crescere della complessità degli elementi). Il modo più semplice é di chiedere tutti gli elementi compatibili con un «elemento campione» di cui alcune parti sono in forma parametrica, ad esempio sostituite dal segno «?».
Così l'elemento (? MILANO ?) é compatibile con gli elementi corrispondenti a tutte le persone nate a Milano, mentre l'elemento (PIETRO ? ?) ricava tutte le persone che si chiamano Pietro.
Il confronto di compatibilità é svolto dal predicato MATCH, che analizza due liste anche strutturate di cui la prima può contenere dei punti interrogativi, e risponde T o NIL a seconda che le due liste siano o no compatibili nel senso visto sopra.
Questo predicato é ora usato nella funzione TROVA1, che costruisce una *sottolista* di una lista L formata da tutti gli elementi di L compatibili con l'elemento parametrico X: la funzione TROVA richiama quest'ultima passandole il nome della banca dei dati, così come ADD richiama INS.
Una ricerca sulla banca di dati può ora essere effettuata nel seguente modo:
(TROVA (? MILANO ?))
e il sistema risponde con una lista contenente gli elementi riguardanti tutte le persone nate a Milano.
Gli elementi possono essere anche *cancellati* dalla banca di dati, singolarmente o in modo parametrico, usando le funzioni DEL1 e DEL, sul cui funzionamento non mi dilungo perché perfettamente analogo alla TROVA1 e TROVA.
Esse fanno si che:
(DEL '(PIETRO ? ?))
provochi la cancellazione dalla banca di dati di tutti gli elementi relativi alle persone di nome Pietro.
L'esempio mostrato nel programma 2 mostra alcune operazioni di inserimento, ricerca e cancellazione su una banca di dati strutturata in questo modo: si ripete però che il formato degli elementi é libero (a parte il nome in prima posizione), e che può essere anche strutturato in sottoliste, ad esempio in questo modo:

(cognome nome professione città
(via numero) (prefisso n° telefonico))

indirizzo *n° telefonico*

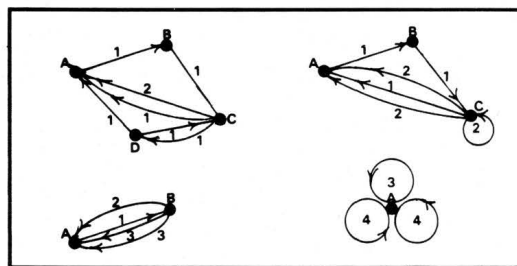
Si noti che una sottolista può essere riassunta con un solo punto interrogativo: ad esempio l'espressione
(? ? INGEGNERE MILANO ? ?)
é compatibile con tutti gli ingegneri milanesi.

Insiemistica, che passione!

Nei due casi discussi sono state presentate due diversissime modalità di impiego delle liste gestite dal GD 010 LISP: sequenza di istruzioni del linguaggio stesso ed elemento di una banca dati; vediamo ora una altro impiego totalmente diverso di questa versatilissima struttura, ossia la lista intesa come *insieme*.
Ormai insegnano insiemistica anche alle elementari, e i concetti di unione, intersezione, appartenenza etc. non dovrebbero avere più segreti. I linguaggi più in uso fino a pochi anni fa (fa eccezione il solo PASCAL, di concezione più recente) non consentivano tuttavia di lavorare sul calcolatore con variabili di tipo «insieme», soprattutto perché la struttura rigida di questi linguaggi non permetteva di manipolare un tipo di dato così variabile. Lo stesso PASCAL necessita di un gruppo di istruzioni appositamente definite per lavorare sugli insie-

Programma n. 4: applicazioni alla teoria dei grafi. Le funzioni di questo programma definiscono un grafo ed eseguono operazioni di cancellazione e di eliminazione di nodi.

```
↑EVALUATE
↑VALUE IS   5

↑EVALUATE
↑VALUE IS   5

↑EVALUATE
↑VALUE IS   NODO

↑EVALUATE   (LAMBDA (X) (SETQ NOD (CONS X NOD)))
↑VALUE IS   (LAMBDA (X) (SETQ NOD (CONS X NOD)))

↑EVALUATE
↑VALUE IS   LATO

↑EVALUATE   (LAMBDA
   (X Y (Z . 1))
   (COND
      ((OR (ABS X NOD) (ABS Y NOD))
         (PRIN0
            (QUOTE NON)
            BLANK
            (QUOTE ACCETTATO)
            CR))
      (T
         (SETQ
            LAT
            (CONS (LIST X Y Z) LAT)))))
↑VALUE IS : )

↑EVALUATE
↑VALUE IS : INCID

↑EVALUATE : (LAMBDA
   (N L)
   (COND
      ((NULL L) NIL)
      ((EQ N (CADAR L))
         (CONS
            (CAR L)
            (INCID N (CDR L))))
      (T (INCID N (CDR L)))))
↑VALUE IS : )

↑EVALUATE
↑VALUE IS : EXCID

↑EVALUATE : (LAMBDA
   (N L)
   (COND
      ((NULL L) NIL)
      ((EQ N (CAAR L))
         (CONS
            (CAR L)
            (EXCID N (CDR L))))
      (T (EXCID N (CDR L)))))
↑VALUE IS : )

↑EVALUATE
↑VALUE IS   SELFLOOP

↑EVALUATE : (LAMBDA
   (L (N . 0))
   (COND
      ((NULL L) NIL)
      ((AND
         (EQL (CAAR L) (CADAR L))
         (OR
            (EQ N 0)
            (EQL N (CAAR L))))
         (CONS
            (CAR L)
            (SELFLOOP (CDR L) N)))
      (T (SELFLOOP (CDR L) N))))
↑VALUE IS : )

↑EVALUATE
↑VALUE IS   CANCEL

↑EVALUATE   (LAMBDA
   (N)
   (SETQ NOD (DEL N NOD))
   (SETQ
      LAT
      (DIFF
         (DIFF LAT (INCID N LAT))
         (EXCID N LAT))))
↑VALUE IS : )

↑EVALUATE
↑VALUE IS   ELIM

↑EVALUATE   (LAMBDA
   (N)
   (SETQ NOD (DEL N NOD))
   (SETQ
      LAT
      (APPEND
         (LATDEL N LAT)
         (RIPRIST
            N
            (DIFF
               LAT
               (SELFLOOP LAT N))))))
↑VALUE IS   )

↑EVALUATE
↑VALUE IS : RIPRIST

↑EVALUATE : (LAMBDA (N L) (RIPR1 N (INCID N L)))
↑VALUE IS : (LAMBDA (N L) (RIPR1 N (INCID N L)))

↑EVALUATE
↑VALUE IS : RIPR1

↑EVALUATE   (LAMBDA
   (N L)
   (COND
      ((NULL L) NIL)
      (T
         (APPEND
            (RIPR2
               (CAR L)
               (EXCID
                  N
                  (DIFF
                     LAT
                     (SELFLOOP LAT N))))
            (RIPRIST N (CDR L))))))
↑VALUE IS : )

↑EVALUATE
↑VALUE IS   RIPR2

↑EVALUATE   (LAMBDA
   (EL LIS)
   (COND
      ((NULL LIS) NIL)
      (T
         (CONS
            (LIST
               (CAR EL)
               (CADAR LIS)
               (PLUS
                  (CADDR EL)
                  (CADDR (CAR LIS))))
            (RIPR2 EL (CDR LIS))))))
↑VALUE IS   )

↑EVALUATE
↑VALUE IS   ESEMPIO

↑EVALUATE
█
↑VALUE IS   (NODO (QUOTE A))

↑EVALUATE
↑VALUE IS   (NODO (QUOTE B))

↑EVALUATE   (B A)
↑VALUE IS   (B A)

↑EVALUATE
↑VALUE IS   (NODO (QUOTE C))

↑EVALUATE   (C B A)
↑VALUE IS   (C B A)

↑EVALUATE
↑VALUE IS   (NODO (QUOTE D))

↑EVALUATE   (D C B A)
↑VALUE IS : (D C B A)

↑EVALUATE
↑VALUE IS   (LATO (QUOTE A) (QUOTE B))

↑EVALUATE   ((A B 1))
↑VALUE IS   ((A B 1))

↑EVALUATE
↑VALUE IS   (LATO (QUOTE B) (QUOTE C))

↑EVALUATE : ((B C 1) (A B 1))
↑VALUE IS   ((B C 1) (A B 1))

↑EVALUATE
↑VALUE IS : (LATO (QUOTE C) (QUOTE D))

↑EVALUATE : ((C D 1)
   (B C 1)
   (A B 1))
↑VALUE IS : )

↑EVALUATE : ((D A 1)
   (C D 1)
   (B C 1)
   (A B 1))
↑VALUE IS : )

↑EVALUATE
↑VALUE IS : (LATO (QUOTE C) (QUOTE A))

↑EVALUATE : ((C A 1)
   (D A 1)
   (C D 1)
   (B C 1)
   (A B 1))
↑VALUE IS : )

↑EVALUATE
↑VALUE IS : (LATO (QUOTE D) (QUOTE C))

↑EVALUATE : ((D C 1)
   (D A 1)
   (C D 1)
   (C A 1)
   (B C 1)
   (A B 1)
   (C A 2))
↑VALUE IS : )

↑EVALUATE :
↑VALUE IS : (ELIM (QUOTE D)) ·

↑EVALUATE : ((C A 1)
   (B C 1)
   (A B 1)
   (C A 2)
   (C C 2)
   (C A 2))
↑VALUE IS : )

↑EVALUATE
↑VALUE IS : (SELFLOOP)

↑EVALUATE :
↑VALUE IS : 5

↑EVALUATE :
█((C C 2))
↑VALUE IS : ((C C 2))

↑EVALUATE :
↑VALUE IS : (ELIM (QUOTE C))

↑EVALUATE : ((A B 1)
   (B A 2)
   (B A 3)
   (B A 3))
↑VALUE IS : )

↑EVALUATE

↑VALUE IS : (ELIM (QUOTE B))

↑EVALUATE : ((A A 3)
   (A A 4)
   (A A 4))
↑VALUE IS : )
```

Fig. 2 - Il grafo descritto dall'esempio del programma 4: vengono successivamente eliminati i nodi D, C e B.

mi, con un grande sforzo a livello di programma traduttore.
Anche qui, invece, il LISP sembra fatto apposta: se noi diamo il nome di «insieme» ad una lista, diventa uno scherzo definire le principali operazioni sulle liste, ed é questa la via seguita dalle funzioni del programma 3. Vediamole una per una:
— appartenenza di un elemento X ad un insieme L: X∈L
la funzione PRES confronta l'elemento x con gli elementi della lista L finché non rova eguaglianza (T) o la lista termina (NIL);
— non appartenenza: X∉L
la funzione ABS é la negazione della precedente;
— unione di due insiemi: A∪B
la funzione UNION costruisce una lista formata da tutti gli elementi di A e da tutti gli elementi di B che non compaiono in A.
Nota: nel caso che una lista sia molto più corta dell'altra, conviene assegnarla al parametro B per ridurre la ricorsività;
— intersezione di due insiemi: A∩B
la funzione INTERS costruisce una lista formata da tutti gli elementi di A che sono presenti anche in B;
— differenza di due insiemi: A-B
la funzione DIFF costruisce una lista formata da tutti gli elementi di A che non sono presenti anche in B.
Come si vede il funzionamento di queste routines é molto semplice e schematico, e risolve con pochi passi di programma e senza alcuno sforzo di compilazione questo spinoso problema.
Siamo ora pronti per un'applicazione dell'insiemistica, e in particolare ho scelto un campo della matematica in cui l'informatica é sempre entrata facendo salti mortali per ridurne le variabili in termini numerici, e non sempre operando in modo efficiente: parlo della *teoria dei grafi*.
Fin dalla definizione di grafo, la teoria viaggia in termini di insiemi e le funzioni del programma 4 vi si attengono pressoché alla lettera, mentre tutte le applicazioni sul calcolatore nei linguaggi convenzionali sono costrette ad usare matrici ed altri elementi aritmetici. Questa mi sembra fra l'altro una dimostrazione di quanto si intende per «intelligenza artificiale» nel caso del LISP.
Vediamo dunque una per una le funzioni del programma 4 ed associamole alle defnizioni date dalla teoria:
— defnizione di grafo: «un grafo é un ente matematico composto di un insieme NOD di elementi detti *nodi* e di un insieme LAT di elementi detti *lati* e formati ciascuno da una triade (X Y P), con X e Y appartenenti all'insieme NOD, e P = funzione peso associata al lato» (ossia (X Y P) significa che un lato con peso P va dal nodo X al nodo Y).
Le funzioni NODO e LATO definiscono un grafo introducendo un elemento per volta: si noti che LATO accetta soltanto coppie di nodi già presenti nell'insieme NOD:
— «si dice insieme dei lati entranti in (uscenti da) un nodo N quel sottoinsieme di LAT formato dagli elementi (X N P)[rispettivamente (N Y P)]».
Le funzioni INCID ed EXCID ricavano questi sottoinsiemi con delle semplici comparazioni su una lista parametrica: al momento della chiamata questa lista deve essere LAT:
— «si dice autoanello un lato del tipo (X X P)» (ossia entrante e uscente dallo stesso nodo).
La funzione SELFLOOP ricava gli autoanelli facenti capo ad un nodo N, o, in mancanza del parametro, tutti gli autoanelli del grafo:
— «si dice cancellazione di un nodo N l'eliminazione dal grafo del nodo N e di tutti i lati entranti ed uscenti da N».
La funzione CANCEL esegue questa operazione sull'insieme NOD e sull'insieme LAT, usando le funzioni INCID ed EXCID e la differenza fra insiemi:
— «si dice eliminazione di un nodo N la cancellazione di tale nodo seguita dall'aggiunta all'insieme LAT di nuovi lati che ripristinano gli eventuali cammini interrotti con la cancellazione di N; ossia, in termini matematici, la aggiunta di tutti i lati del tipo (X Y P1+P2) quando esistessero, prima della cancellazione, i lati (X N P1) ed (N Y P2)».
La funzione ELIM esegue questa operazione con l'aiuto delle tre sottofunzioni RIPRIST, RIPR1 e RIPR2. È forse più chiaro spiegare il suo funzionamento con l'esempio riportato nel programma 4, che fa riferimento alla fig. 2).
In esso un grafo formato da quattro nodi e sei lati con peso unitario viene progressivamente ridotto mediante l'eliminazione dei nodi D, C e B: gli autoanelli presenti nel nodo finale A rappresentano tutti i cicli del grafo originario che passavano per questo nodo, e il peso indica la loro lunghezza (ossia da quanti lati erano formati).

Conclusione

Gli esempi esposti, pur se incompleti, dimostrano una volta di più le sconfinate possibilità del LISP e dell'intelligenza artificale nel campo dei piccoli calcolatori. Finalmente abbiamo a disposizione uno strumento per sviluppare questa ancora tanto sconosciuta branca dell'informatica, ed é auspicabile che anche altri costruttori si muovano su questa strada. Le applicazioni immediate sono molteplici, da una gestione svelta ed essenziale degli archivi alla definizione di linguaggi a livello superiore che si «appoggiano» sul LISP (ne é un esempio quello sviluppato per lavorare sugli insiemi).
Oltre a ciò, oltre cioé alla pura convenienza, l'uso del LISP é consigliabile a chiunque voglia astrarsi un po' dalla routine de «conti della spesa» più o meno elaborati e lavorare con una logica a più alto livello, se non altro per avere un rapporto più costruttivo con i nostri strumenti di lavoro.

Pietro Hasenmajer

Ecco l'indice delle 16 biblioteche «UNIVERS»

1 - **CIVIL 1:** Telaio ortogonale a nodi spostabili, galleria circolare, bilanciamento reti idriche, travi in c.a.p., poligonale, verifica a presso-tensoflessione, palo in terreno multistrato, muro di sostegno, trave continua, cremoniano, trave su suolo elastico.
2 - **VORTEL:** Determinazione delle forze orizzontali sismiche (normativa italiana). Telaio piano-ortogonale ad aste con inerzia costante. Telaio piano-ortogonale ad aste con variazione d'inerzia, comunque vincolate a terra fra asta ed asta.
3 - **VINTEL:** Analisi dinamica e calcolo di telai piani generici (aste comunque inclinate e comunque vincolate, aste con variazione d'inerzia lineare e/o parabolica, cedimenti, distorsioni, variazioni, temperature, ecc.).
4 - **TRAVEL:** Travi continue su suolo elastico e linee d'influenza delle azioni esterne mobili.
5 - **GESTR:** Travi continue, diagrammi inviluppo per travate continue, travi in c.a.p., strutture reticolari, strutture di fondazione, solaio continuo, impalcato più spalla, tombino scatolare.
6 - **GEVER:** Progetto e verifica di sezioni in c.a.p., poligonali, circolari e a T; diagrammi di plasticizzazione (D.P.R. 26/3/80).
7 - **STEND:** Verifica di stabilità dei pendii in condizioni piane calcolata col metodo dell'equilibrio allo stato limite.
8 - **ERPS:** Progetto stradale completo. Geometrizzazione del tracciato plano-altimetrico - Progetto e calcolo delle sezioni stradali - Movimenti di terra - Disegno del tracciato, dei profili e delle sezioni stradali.
9 - **IDRAL:** Progetto completo di reti idriche e fognature: calcoli, relazioni tecniche, disegni esecutivi.
10 - **TERMIN:** Impianti di riscaldamento (legge 373), di condizionamento e impianti frigoriferi.
11 - **DISFER:** Calcolo, disegno e computo metrico del ferro per strutture in c.a.
12 - **PROSPE:** Costruzione e restituzione dell'immagine, restituzione prospettica.
13 - **TOTOR:** Sistemi totocalcio ridotti con metodi statistici.
14 - **CONGE:** Contabilità generale e contabilità IVA.
15 - **PAGHE:** Gestione del personale, paghe e stipendi.
16 - **GEMAG:** Contabilità di magazzino, contabilità clienti, fatturazione.
17 - **PRECOL:** Analisi e revisione prezzi. Contabilità dei lavori. Stati di avanzamento. Computi metrici. Offerte appalti concorso.
IN PREPARAZIONE: Risoluzione dei problemi elastici con il metodo degli elementi finiti.

NOME E COGNOME
TELEFONO
SOCIETÀ/ENTE
INDIRIZZO
PROFESSIONE
SONO INTERESSATO ALLE APPLICAZIONI DI
◇ CALCOLO STRUTTURE
◇ TOPOGRAFIA COSTRUZIONI DI STRADE
◇ PLOTTAGGIO DISEGNI ESECUTIVI
◇ RELAZIONI TECNICHE
◇ CONTABILITÀ GENERALE, I.V.A., PAGHE
◇ CONTABILITÀ DEI LAVORI, COMPUTI METRICI, CAPITOLATI APPALTI
◇ ALTRO

commodore

N°1 IN MICROCOMPUTERS

Apparecchiature originali e compatte costruite con altissima tecnologia.
Una vastissima rete di distribuzione ed assistenza tecnica.
Un servizio programmi di alta professionalità con coordinamento ed apporti a livello mondiale-europeo-italiano.
Hardware e Software orientati ad un uso facile e sicuro per l'utente.
Investimenti adeguati ed a lungo periodo.

Queste sono alcune delle ragioni che hanno decretato il larghissimo consenso degli utilizzatori, tanto da rendere i computer ed i sistemi HARDEN Commodore i più venduti in Italia.

ORGANIZZAZIONE UFFICIALE COMPUTERS
COMMODORE
CONCESSIONARI REGIONALI:
PIEMONTE: ABA ELETTRONICA (011/501512)
LIGURIA: PIRISI (0185/301032)
LOMBARDIA: HOMIC (02/4695467)
TRENTINO ALTO ADIGE: WIKUT (0472/21552)
TREVISO E BELLUNO: COREL (0432/291466)
FRIULI VENEZIA GIULIA: ELMA ELETTRONICA (040/793211)
VENETO (ESC. TV E BL.): H.S.H. (0445/43061)
EMILIA ROMAGNA: SHR (0544/30258)
TOSCANA: MCS (055/571380)
UMBRIA: ATLAS SYSTEM (0761/224688)
MARCHE ABRUZZI E MOLISE: INFORAB (085/31653)
LAZIO: S.I.L. (0773/43771)
CAMPANIA: MEG SYSTEM (081/261344)
PUGLIE E BASILICATA: BAS (0881/76111)-(080/227575)
CALABRIA: SIRANGELO (0984/71392)
SICILIA: EDIL COMPUT (090/2928269)
SARDEGNA: SII INFORMATICA (070/663746)
DISTRIBUTORI AUTORIZZATI IN TUTTE LE PROVINCIE ITALIANE

HARDEN

PER L'ITALIA:
HARDEN S.p.A.
26048 SOSPIRO (Cremona)
Tel. 0372/63136 r.a.-Telex 320588

la divisione in sil-la-be

Contrariamente a quanto spesso si creda, nella lingua italiana la divisione delle parole in sillabe è soggetta a regole ben precise. È possibile realizzare programmi (abbastanza, ma non «tragicamente», complessi) che consentano al calcolatore di eseguire corrette divisioni. Una routine di questo genere è particolarmente adatta ad essere inserita in programmi per il trattamento della parola.

Recentemente ho avuto occasione di scrivere un *word processor* in BASIC per un calcolatore IBM 5110. Avevo intenzione di includervi una routine per la divisione delle parole in sillabe, in modo da poter introdurre il testo come un'unica lunghissima stringa. La stampa sarebbe poi potuta avvenire su un numero di colonne di volta in volta variabile (cioè, come si dice, con *giustificazioni* diverse) e il programma avrebbe provveduto automaticamente alla divisione delle parole e all'inserimento del trattino di a-capo.

Per documentarmi, ho cominciato con il controllare alcuni stampati ottenuti con *word processor*, ma nessuno di quelli che mi sono capitati sottomano presentava divisione delle parole. Ogni riga terminava con una parola completa, e se vi era l'incolonnamento a destra, questo era ottenuto inserendo ulteriori spazi bianchi all'interno della riga.

Ho svolto una piccola indagine anche sulle macchine fotocompositrici (che sono particolari calcolatori dedicati, nei quali gira un programma per il trattamento dei testi), e ho notato che anche in alcune di queste la divisione della parola non avveniva automaticamente, ma doveva essere fatta dall'operatore, e quindi su una lunghezza di riga (*giustezza*) prestabilita.

Mi è poi capitato di leggere il libro di D. E. Knuth, il noto informatico americano, autore dei volumi *The art of computer programming*. Il libro in questione è *TEX and Metafont, New Directions in Typesetting* e vi viene descritto un programma di preparazione dei testi per la fotocomposizione. Qui la divisione in sillabe (in inglese, *hyphenation*) c'è. Tuttavia, leggendo il capitolo dedicato all'argomento, mi sono reso conto che il metodo non è adatto per l'applicazione che mi interessa. Anzitutto, perché si riferisce alle parole della lingua inglese, che hanno un sistema di divisione non applicabile alla nostra lingua. Poi, perché il metodo è particolarmente complesso richiedendo, tra l'altro, la registrazione su disco di una serie di eccezioni (circa 350 parole) alle varie regole. La complessità del metodo deriva dal fatto che l'*hyphenation* dei termini inglesi segue regole molto complicate: per esempio *record* si divide in *rec-ord* se è un sostantivo e in *re-cord* se è un verbo. E ancora, il termine *hyphenation* stesso si divide in *hy-phen-a-tion*, ma *concatenation* in *con-cat-e-na-tion*.

Bene; ma per l'italiano, le cose come stanno? Esiste una regola ragionevolmente applicabile, o, come per l'inglese, la questione è troppo complessa?

Consultando alcune grammatiche italiane (tra cui S. BATTAGLIA e V. PERNICONE *La grammatica italiana* Loescher, 1977) mi sono convinto che le regole non sono così complicate, ed ho cercato di esprimerle in modo orientato all'algoritmo. Ci sono tre regole principali:

1. Se c'è una successione di vocali e consonanti (vocale, consonante, vocale, consonante,... oppure consonante, vocale, consonante, vocale,...) dividere prima delle consonanti; per esempio
de-li-ca-to
e-di-fi-ca-re.

2. Un gruppo consonantico (due o più consonanti) nella maggioranza dei casi si divide

```
0001 REM ************************************
0002 REM *                                  *
0003 REM *   PROGRAMMA PER DIVIDERE UNA     *
0004 REM *        PAROLA IN SILLABE         *
0005 REM *         (MAURO BOSCAROL)         *
0006 REM *                                  *
0007 REM ************************************
0010 INPUT A$
0020 IF A$<>"*" GOTO 40
0030 STOP
0040 H=LEN(A$)
0050 IF H<=2 GOTO 370
0060 I=1
0070 GOSUB 1000
0080 IF L=0 GOTO 340
0090 L1=L
0100 I=I+1
0110 IF I>H GOTO 370
0120 IF MID$(A$,I,1)="H" GOTO 100
0130 GOSUB 1000
0140 IF L=0 GOTO 250
0150 IF L1=2 OR L=2 GOTO 190
0160 J=I-1
0170 GOSUB 2000
0180 GOTO 40
0190 I=I+1
0200 IF I>H GOTO 370
0210 L1=L
0220 GOTO 130
0250 I=I+1
0260 IF I>H GOTO 370
0270 GOSUB 1000
0280 IF L=0 GOTO 310
0290 J=I-2
0300 GOTO 170
0310 GOSUB 3000
0320 IF S=0 GOTO 160
0330 GOTO 290
0340 I=I+1
0350 IF I>H GOTO 370
0360 GOTO 70
0370 PRINT A$
0380 GOTO 10
1000 REM VOCALE O CONSUNANTE?
1010 L$=MID$(A$,I,1)
1020 IF L$="I" OR L$="U" GOTO 1080
1030 IF L$="A" OR L$="E" OR L$="O" GOTO 1060
1040 L=0
1050 RETURN
1060 L=1
1070 RETURN
1080 L=2
1090 RETURN
2000 REM STAMPA DELLA SILLABA
2010 PRINT MID$(A$,1,J);"-";
2020 A$=MID$(A$,J+1)
2030 RETURN
3000 REM GRUPPO CONSUNANTICO
3010 L$=MID$(A$,I-1,1)
3020 M$=MID$(A$,I,1)
3030 IF L$=M$ GOTO 3090
3040 IF L$="G" AND M$="N" GOTO 3110
3050 IF L$="L" OR L$="M" OR L$="N"
     OR L$="R" OR L$="V" OR L$="Z" GOTO 3090
3060 IF L$="S" OR M$="R" GOTO 3110
3070 IF L$="T" GOTO 3090
3080 IF M$="H" OR M$="L" GOTO 3110
3090 S=0
3100 RETURN
3110 S=1
3120 RETURN
```

Listing del programma per dividere una parola in sillabe. Può essere eseguito su Apple II, PET, TRS-80.

dopo la prima consonante (sempre, quando le prime due consonanti del gruppo sono uguali). Si ha eccezione a questa regola, e la divisione va messa prima del gruppo consonantico, quando la prima consonante del gruppo è una *s* (si tratta di s impura, cioè seguita da una consonante) e negli altri casi elencati nella tabella 1. Per esempio

ap-pret-to	(*pp* e *tt* si separano)
ar-tri-te	(*rt* si separa)
di-ste-so	(*st* resta unito)
ec-to-pla-sma	(*pl* e *sm* restano uniti).

3. Se c'è un gruppo vocalico (due o più vocali), può trattarsi di un gruppo divisibile o indivisibile (detto dittongo o trittongo, ecc.) secondo il numero di vocali di cui è composto (si veda la tabella 2 e la prosecuzione del testo per le definizioni di dittongo e trittongo). Per esempio

be-a-to	(*ea* non è mai dittongo)
ma-e-stro	(*ae* non è mai dittongo)
mi-o	(*io* in questo caso non è dittongo perché *i* porta l'accento)
mie-le	(*ie* in questo caso è dittongo perché l'accento è sulla e)
pi-gliai	(*iai* è trittongo: accento su *a*)
miei	(*iei* è trittongo: accento su e)

Le regole 1 e 2 possono essere inserite in un programma. Purtroppo la regola 3 è più complicata (non solo per un programma, ma anche per un essere umano). La complicazione sta nel decidere se due (o tre) vocali formano dittongo (o trittongo).

Un dittongo è una coppia di vocali, di cui una è *u* o *i* non accentata. Come fa il programma a sapere quando la *u* o la *i* è accentata? Naturalmente non può. Si badi che l'accento di cui si parla non è quello «grafico», cioè quello che si segna sulle lettere, ma quello «tonico», cioè quello che caratterizza l'intonazione della parola (per esempio, in *nave* l'accento tonico è sulla *a*).

Seguendo la regola, si può dire che le due vocali *i* e *a* in *fiamma* formano dittongo, perché una delle due è una *i* non accentata. Il dittongo non si può separare, e così la divisione della parola è *fiam-ma*. Invece nella parola *paura*, la coppia di vocali *a* e *u* non forma

```
A-E-RO-NAU-TI-CA
RE-STAR-SE-NE
A-GNO-STI-CO
MA-E-STRO
CUO-RE
AC-QUI-REN-TE
IP-NO-SI
MAN-GIA-RE
OC-TO-PLA-SMA
IP-SI-LON
AN-GLI-CA-NO
AR-TRI-TE
MIO
MIEI
```

Tabella 3. Divisione in sillabe di alcune parole effettuata con il programma descritto in questo articolo. Le divisioni indicate sono tutte corrette. Il programma non riconosce la divisione mi-o perché non riesce a decidere se io è dittongo oppure no (in altre parole, non riesce a distinguere tra mìo e miò).

PROGETTI PER MIGLIORARE IL PROGRAMMA

Nei casi in cui l'ho sperimentato, questo programma ha dato divisioni esatte delle parole. Possono tuttavia esistere delle parole particolarmente complesse la cui divisione non viene data correttamente. Solo l'inserimento del programma in un *word processor* ed il suo uso prolungato potranno dimostrarne la correttezza, o suggerire le opportune modifiche.
Altre modifiche si possono fare per comprendere parole straniere entrate nell'uso corrente. Anzitutto bisognerà considerare il caso delle lettere *x, y, w, j, k* di cui alcune andranno considerate come vocali e altre come consonanti. Quindi bisognerà studiare la divisione di parole come: standard, fan, snob, beat, transistor, cocktail.
Altre modifiche si possono fare per il riconoscimento dei dittonghi: per esempio nelle parole in cui *i* è preceduta da *r* o da un gruppo consonantico, non si ha dittongo: *ri-a-pri-re, ri-ot-ten-go, o-ri-en-te, ri-o-ne.*
Bisognerà poi prevedere la presenza della punteggiatura: apostrofo, punto, virgola, due punti, virgolette, parentesi.
Infine si potranno esaminare le sigle: kHz, mWb, ecc.

dittongo perché la *u* è accentata: *pa-u-ra*. Infine nella parola *maestro* la coppia di vocali *a* e *e* non fa dittongo perché non compare una *u* né una *i: ma-e-stro*.
Analogamente, il trittongo è costituito da tre vocali, di cui due sono *u* e/o *i* non accentate e la terza è qualunque. Per esempio le tre vocali di *buoi* fanno trittongo (qui l'accento è sulla *o*), e dunque questa parola non si può dividere.
Esistono poi pochissime parole con un gruppo di quattro vocali, come *aiuola* e *cuoio*.
Se dunque la decisione sui dittonghi e trittonghi dipende da un accento (tonico), il programma non potrà proprio prenderla. Nonostante ciò, qualcosa (anzi, molto) ancora si può fare. Anzitutto perché i problemi riguardano solo le parole con dittonghi e trittonghi, che non sono poi moltissime. Poi, perché il fatto di non poter talvolta decidere sui dittonghi si

Un dittongo è formato da una di queste coppie di vocali
ia ie io
ai ei oi
ua ue uo
au eu ou
a condizione che la *i* o la *u* non portino l'accento; oppure da una coppia
ui iu
potendo in tal caso una delle vocali essere accentata.
Un trittongo è costituito da una delle terne di vocali seguenti
uxu ixi
xui uxi uix
xiu ixu iux
dove x è una delle vocali *a, e, o* a condizione che *i* e *u* non portino l'accento.

Figura 1. Algoritmo per trovare la prima sillaba di una parola. Una volta determinata, la prima sillaba viene eliminata e l'algoritmo si riapplica.

	b	c	d	f	g	h	l	m	n	p	q	r	s	t	v	z
b							bl					br				
c						ch	cl					cr				
d												dr				
f												fr				
g						gh	gl		gn			gr				
h																
l																
m																
n																
p							pl					pr				
q																
r																
s	sb	sc	sd	sf	sg		sl	sm	sn	sp	sq	sr		st	sv	sz
t												tr				
v																
z																

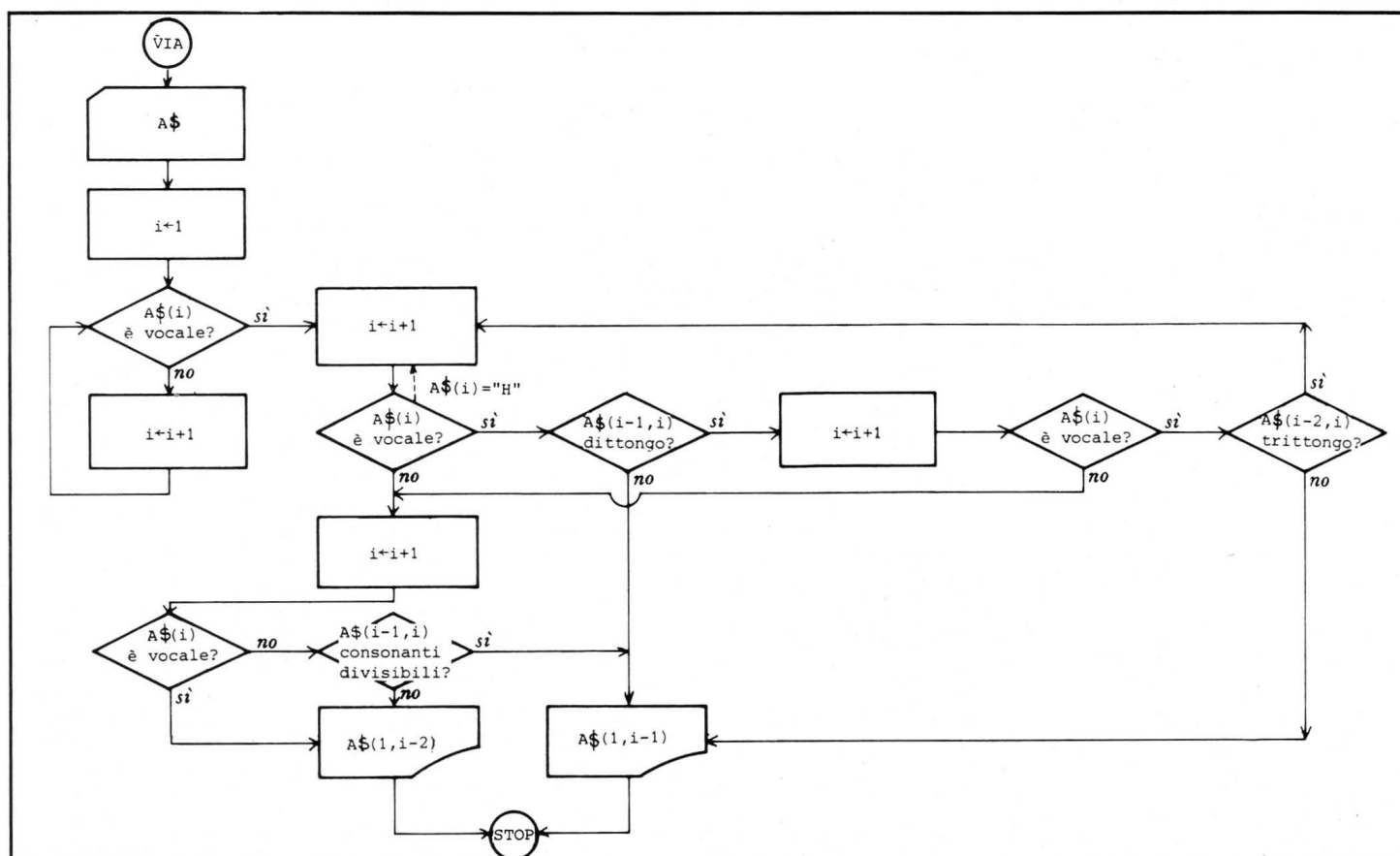



Tabella 1. I gruppi consonantici che iniziano con una coppia di consonanti riportata in tabella non si dividono (cioè la divisione va messa prima del gruppo consonantico). Tutti gli altri gruppi si dividono dopo la prima consonante del gruppo. (Le coppie di consonanti riportate in tabella sono quelle con le quali può iniziare una parola della lingua italiana).

traduce nel non mettere un trattino di separazione in un posto in cui forse andrebbe, cioè nel riconoscere meno separazioni di quelle che ci sono. Non è poi così grave. Sarebbe certo peggio trovare separazioni inesistenti. A questo proposito il Gabrielli a pag. 36 del suo libro *Si dice o non si dice?* pubblicato da Mondadori, riscontrando la difficoltà di riconoscimento dei dittonghi, consiglia di non separare mai due vocali. Questo mi sembra troppo drastico, perché, almeno nei casi in cui compare né *u* né *i*, la separazione è certa e facilmente riconoscibile.

Mettendo assieme queste regole, si può sviluppare un algoritmo per risolvere il problema della divisione in sillabe delle parole della lingua italiana. In figura 1 è riportato un diagramma di flusso dell'algoritmo. Questo algoritmo non tiene conto delle lettere speciali (non dell'alfabeto italiano) *j*, *k*, *w*, *x* e *y*. Inoltre qui per «dittongo» si intende: coppia di vocali, di cui una *i* o *u*; e analogamente per «trittongo».

Inizialmente viene introdotta la parola, che qui è indicata con A$ in analogia con il BASIC. Con la notazione A$(i) si intende l'i-esimo carattere di A$, mentre A$(i,j) indica i caratteri di A$ dall'i-esimo al j-esimo. L'algoritmo trova la prima sillaba della parola. Appena trovata, questa viene eliminata dalla parola e la procedura si riapplica.

L'algoritmo inizia con il cercare la prima vocale. Dopo la prima vocale può esserci una consonante o una vocale. Nel primo caso ** si può trattare di una consonante semplice (cioè seguita da vocale) ed allora la divisione avviene alla sua sinistra, o di un gruppo consonantico, che può essere divisibile (dopo la prima consonante) oppure no (divisione prima della prima consonante) secondo le regole riportate in tabella 1 e codificate nella subroutine che inizia all'istruzione 3000.

Nel secondo caso *, le due vocali possono non formare dittongo, ed allora la divisione avviene tra loro, oppure possono formare dittongo. In quest'ultimo caso si passa ad esaminare la prossima lettera: se è una vocale si torna a *; se è una consonante si passa a **.

Questo algoritmo è stato codificato in un programma BASIC che può essere eseguito indifferentemente su PET, TRS-80 e Apple II. Con la sostituzione

MID$(A$, i, j)→SEG$(A$, i, i+j-1)

può essere eseguito su PDP-11 e su tutti i calcolatori che hanno la funzione SEG$ invece della MID$.

In conclusione, questa routine è abbastanza breve da poter essere inserita in un word processor scritto per personal computer, ottenendo così una divisione automatica delle parole che devono essere spezzate per andare a capo. Il procedimento potrebbe essere questo. Inserire il testo come vettore di stringhe lunghe, per esempio, 255 caratteri. Quando si vuole la stampa, introdurre il numero di colonne n per l'impaginazione del testo. Tagliare la prima stringa a n colonne. Se il taglio capita a metà parola, trovare la divisione in sillabe della parola, e prendere la divisione all'immediata sinistra del taglio. Aggiungere un trattino di a-capo e incolonnare a destra. Naturalmente, è vantaggioso trascrivere il programma in linguaggio macchina, per aumentare la velocità di esecuzione.

Mauro Boscarol

SHARP MZ - 80 K

Lo splendido personal che viene dal Giappone

INTERNORD

elegante, versatile, espandibile

Lo Sharp MZ-80 K è il primo personal giapponese che entra sul mercato mondiale.

Guardatelo: si capisce immediatamente che è stato studiato già oggi per l'impiego di domani; la sua estetica rivela infatti funzionalità e tuttavia eleganza, compattezza, leggerezza.

Il suo campo di applicazione è vastissimo: parte dai giochi più sofisticati, divertenti e impegnativi e diventa una valida banca di dati e un perfetto ausilio per il calcolo tecnico e scientifico.

L'hardware è ricco: RAM da 20 Kbyte espandibile a 48 Kbyte, memoria di massa ed audiocassette, video b/n da 40 caratteri per 25 righe, tastiera QWERTY con sezione grafica o tastierina numerica in alternativa.

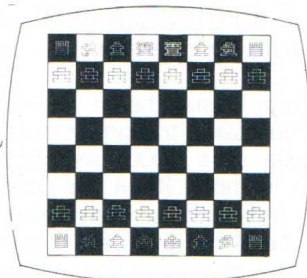
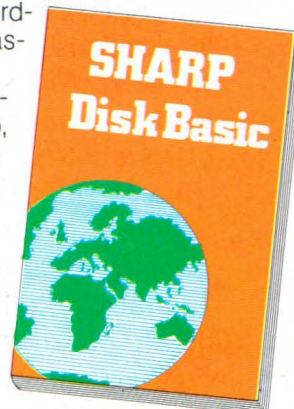
Software: BASIC standard molto veloce dotato di istruzioni per il tracciamento e per la musica (lo MZ-80K è dotato di output acustico modulabile in tono e durata).

Lo Sharp MZ-80 K offre inoltre la massima possibilità di espansione.

Affiancato dalla memoria di massa a minidischi Sharp MZ-80 FD (fino a due unità) da 2 minidischi ciascuna per un totale di 560 Kbyte con tempo di accesso di 20 msec) e dalla stampante Sharp MZ-80 P3 da 80 colonne, 1, 2 righe secondo, lo MZ-80 K diventa un eccellente minisistema in grado di svolgere il lavoro di contabilità generale, IVA, fatturazione, gestione magazzino, di una piccola azienda, di un albergo, di un ristorante, di un negozio (lo MZ-80 K può essere interfacciato con il registratore di cassa Sharp). Lo Sharp MZ-80 K è l'unico personal oggi sul mercato italiano a essere dotato di un manuale BASIC in italiano che spiega in modo semplice questo utilissimo software. Aspetto elegante, hardware affidabile, software ai massimi della categoria.

E il servizio? Per quanto riguarda questo ultimo aspetto, così importante da diventare addirittura vitale per un elaboratore, ci limiteremo a dire che lo Sharp è distribuito dalla Melchioni Computertime che mette a sua disposizione il suo efficiente servizio di consulenza e di assistenza.

@ prove #

Il PET continua a crescere. Prima è cresciuta la tastiera, e dal 2001 si è passati al 3032. Ora è la volta del video, che nell'8032 è da 80 colonne. Ma anche il BASIC è migliorato, e anche la gestione video, e anche il DOS. Eppure, questa interessante macchina ha impiegato più di sei mesi per arrivare da Hannover, dove è stata presentata, in Italia.

Dell'8032, il «PET grande» con video da 80 colonne, si parla ormai da parecchio tempo. Lo avevamo annunciato già nella descrizione del CBM 3032, sul numero 3 di m&p COMPUTER circa dieci mesi fa, nel febbraio di quest'anno. L'8032 è stato presentato alla Fiera di Hannover, in maggio. E in Italia per parecchio tempo non è arrivato: o, meglio, è arrivato ma per alcuni mesi non lo si è venduto nell'attesa di mettere a punto una politica commerciale ed una sufficiente quantità di software per supportarlo. La macchina, però, è valida e desta interesse: visto che l'importatore ufficiale non ne iniziava la commercializzazione, qualcuno ha allora pensato di importare direttamente (e vendere) l'8032. Lo abbiamo riferito in una notizia nel numero precedente: la Deniel's di Torino ha iniziato la vendita di questa macchina, publicizzandola contemporaneamente sulla nostra rivista. Non potevamo non interessarci di... una macchina così interessante, per usare un giro di parole, se non altro per dovere di informazione: prescindendo, ovviamente, dalle politiche commerciali. Così, è stata la Deniel's che, dietro nostra richiesta, ci ha messo a disposizione l'apparecchio per la redazione di questo articolo.
Nel frattempo, ed in extremis per la pubblicazione su questo numero, giunge in redazione (25 novembre, ore 10 e 42) un telex della Harden che annuncia l'inizio della distribuzione, comunicando i prezzi ufficiali della nuova serie. Se questo fosse un «libro giallo», diremmo che il caso può considerarsi chiuso. Ma rimane almeno il rammarico delle polemiche che, per forza di cose, hanno accompagnato lo svolgersi di questa operazione.

Il sistema Commodore

L'8032 è attualmente il modello di punta della gamma di personal computer Commodore, costituita da quattro serie: la serie 2001, la serie 3000, la 4000 e la 8000. Alla serie 2001 appartiene il «vecchio» PET (quello con i tasti piccoli, per intenderci), nelle versioni da 8, 16 e 32 K byte; della serie 3000 fa invece parte il modello descritto nel numero 3 di m&p COMPUTER, il 3032 (disponibile in versione da 16 o da 32 K), con video verde e tastiera «normale», che è rimasto fino a questo momento il modello più evoluto della gamma. Le novità sono le ultime due serie, la 4000 e la 8000: la serie 4000 differisce dalla 3000 per l'adozione di una nuova versione del BASIC, la 4.0 al posto della 3.0; la memoria può essere da 8, 16 o 32 K. Infine, alla serie 8000 appartiene l'8032, descritto in queste pagine: differisce dai modelli della serie 4000 per il video, che è da 80 colonne anziché da 40; il BASIC è sempre il 4.0 e la capacità di memoria è di 32 K byte, non esistono le versioni inferiori.
È importante sottolineare che un modello della serie 3000 può essere trasformato nel corrispondente della serie 4000, semplicemente sostituendo la memoria ROM che contiene il BASIC, con indiscutibile convenienza dal punto di vista delle possibilità operative: naturalmente non altrettanto può avvenire per passare dalla serie 4000 alla 8000; l'unica diversità fra queste due, ripetiamo, consiste nel raddoppiamento della capacità del video della seconda rispetto alla prima.
Per le unità floppy il discorso è in parte

analogo: i modelli 2040/3040 con due dischi da 5 pollici e un quarto della capacità di 170.180 byte ciascuno e con DOS versione 1 (vedi m&p COMPUTER n. 3), sono stati affiancati dall'unità 4040 (sempre con due floppy a singola faccia, ma con nuovo DOS versione 2.0 e un lieve incremento della capacità, salita a 174.848 byte per disco) e dalla 8050, dotata di DOS versione 2.5 e della notevole capacità di 533.248 byte per ognuno dei due dischi (sempre a singola faccia): in totale, circa un mega in linea, una prestazione tutt'altro che da sottovalutare per un personal computer. Come per l'unità centrale, la sostituzione della ROM di sistema operativo può consentire alla 3040 l'adeguamento del DOS e, quindi, il «passaggio» di categoria.
Veniamo alle stampanti: rimangono le 3022 e 3023, di derivazione Epson, rispettivamente con e senza cingoli per il trascinamento della carta (vedi m&p COMPUTER n. 3); ad esse si affianca un modello Tally (che, «vestito» dalla Commodore, ha assunto il nome di CBM 8024) capace di stampa bidirezionale di righe fino a 132 caratteri, alla velocità di 160 caratteri al secondo.
I programmi Commodore per il futuro prevedono, a breve scadenza, due unità floppy da 8 pollici (doppio floppy, doppia densità, singola o doppia faccia, per un totale di 1.5 o 3 megabyte); seguirà, pare, un disco rigido. Da tempo è inoltre allo studio, come abbiamo accennato quasi un anno fa nella presentazione del 3032, un modello con video a colori. Del VIC 1001, la nuova macchina eccezionalmente economica, riferiamo nelle notizie. Per questo prodotto c'è da aspettarsi un grosso successo, anche superiore a quello degli altri modelli, specie presso i giovani e gli hobbysti in genere. Bisogna solo sperare che le novità Commodore non debbano essere soggette ad una lunga anticamera sui mercati esteri prima di riuscire ad «approdare» anche in Italia; questo discorso ha, purtroppo, uno sgradevole precedente proprio nell'8032, giunto in Italia ad oltre sei mesi dalla sua presentazione ad Hannover.

Descrizione

L'8032 si presenta, esteticamente, molto simile ai modelli della serie 3000. L'unica differenza evidente è costituita dal monitor: la dimensione dello schermo è aumentata, infatti, passando da 9 a 12 pollici (cioè da 22.5 a 30 centimetri di diagonale), per poter ospitare il maggior numero di caratteri che possono essere visualizzati. Ricordiamo che la serie 3000 ha un video da 25 righe di 40 caratteri ciascuna, per un totale di 1000 caratteri, mentre nell'8032 le righe sono sempre 25 ma le colonne sono 80: in tutto, duemila caratteri. Prescindendo dall'indiscutibile vantaggio dovuto alla maggior capacità del video, riteniamo che il sistema ne guadagni anche da un punto di vista estetico, grazie al maggior equilibrio delle masse del video e della parte inferiore del contenitore con la tastiera. A proposito di quest'ultima, si nota qualche diversità rispetto a quella della serie 3000. Sui tasti non sono riportati i simboli grafici, mentre la fila superiore serve per la scrittura sia dei numeri sia, premendo il tasto SHIFT, dei consueti simboli (virgolette, dollaro, parentesi eccetera): come nella maggioranza dei calcolatori, ed inversamente a quanto avviene di regola nelle macchine per scrivere. I numeri possono indifferentemente essere introdotti per mezzo della tastiera principale o del tastierino numerico posto sul lato destro: a proposito di quest'ultimo segnaliamo positivamente che l'effetto dei tasti non cambia se contemporaneamente viene premuto lo SHIFT della tastiera principale, il che non sempre avviene nei personal computer. Oltre ai tasti corrispondenti ai normali caratteri ASCII, ve ne sono altri che servono per funzioni speciali e per il controllo del cursore. Il TAB produce una tabulazione analoga a quella delle macchine per scrivere (SHIFT-TAB serve per fissare una tabulazione nella posizione in cui è il cursore, o per cancellarla se già precedentemente fissata). L'ESC serve, fondamentalmente, per uscire dal «modo virgolette»: quando si aprono le virgolette, infatti, i tasti di controllo cursore sono disattivi e vengono codificati per poter essere inseriti in uno statement di PRINT; in caso di errore, dunque, non è possibile tornare indietro agendo sul cursore (ma continua a funzionare il DELETE), a meno di non chiudere le virgolette. La pressione del tasto ESC consente, ora, di recuperare immediatamente il controllo del cursore. Il tasto RVS serve, come nei modelli precedenti, per selezionare la scrittura in REVERSE, dalla quale si esce premendo di nuovo il tasto e contemporaneamente lo SHIFT; come il controllo del cursore, anche questo comando viene codificato se eseguito dopo l'apertura di virgolette. Come nella serie 3000 è presente il tasto HOME/CLR, che ha l'effetto di riportare il cursore in alto a sinistra sullo schermo, cancellando qualsiasi indicazione se premuto contemporaneamente allo SHIFT: anche questo comando può essere inserito, fra virgolette, in uno statement di PRINT. Il tasto DEL/INST (DELETE/INSERT), come nei modelli precedenti, provoca lo spostamento del cursore verso sinistra e la cancellazione dei caratteri incontrati; lo SHIFT ne muta il funzionamento, e la parte di testo che segue la posizione del cursore viene spostata verso destra per far posto ai caratteri da inserire. L'INSERT è programmabile, quindi non attivo dopo le virgolette aperte, il DELETE no.
Il tasto REPEAT, del quale apprezziamo molto la presenza, può essere premuto contemporaneamente a qualsiasi altro, provocando la ripetizione del corrispondente carattere: se ad esempio si premono contemporaneamente l'asterisco e il REPEAT, sullo schermo verrà scritta una successione di asterischi fin quando non si rilascerà almeno uno dei tasti. Questa funzione è utile tutte le volte che è necessario introdurre una sequenza di caratteri uguali. Alcuni tasti sono provvisti di repeat automatico, vale a dire che basta continuare a tenerli premuti per provocare la ripetizione del carattere o del comando: sono il DELETE/INSERT e i tasti per il controllo del cursore. Con questo viene colmata una lacuna delle precedenti serie Commodore, cioè l'assenza assoluta della funzione di ripetizione: il fatto,

Costruttore:
CBM - Commodore Business Machines Divisione della Commodore International Limited 3300 Scott Blvd Santa Clara, California 95050 U.S.A.

Distributore per l'Italia:
Harden - Sospiro (Cremona)

Prezzi:
CBM 8032: L. 2.299.500.
CBM 8032 + doppio floppy 1 M byte CBM 8050 + stampante 132 col. 160 cps CBM 8024 L. 8.276.000.

Riferimento servizio lettori: 32

Ringraziamo la Deniel's (Via Paolini 18 - 10138 Torino) per aver messo a disposizione l'apparecchio per la redazione dell'articolo.

poi, di dotare alcuni tasti di repeat automatico e di prevedere un comando per la ripetizione di qualsiasi altro carattere ci sembra degno del massimo apprezzamento, nel senso che non vi poteva essere una scelta migliore. A parte il classico RETURN, restano i due tasti che consentono il movimento del cursore (con repeat automatico, come abbiamo detto): uno serve per gli spostamenti in giù o (SHIFT) in sù, l'altro per gli spostamenti verso destra o (SHIFT) sinistra; confermiamo a questo proposito la critica mossa al 3032, cioè che sarebbe stato preferibile adottare quattro tasti separati: a maggior ragione per l'8032 che, quanto a gestione del video, è per altri versi decisamente ben dotato. Ciò che non abbiamo assolutamente apprezzato è il funzionamento dello SHIFT LOCK: non si limita, come consueto, ad agire sulla funzione dei tasti di carattere, ma ha effetto sia su questi, sia sui numeri (non quelli del tastierino numerico), sia sui tasti di controllo cursore, sul DELETE, che diviene INSERT, sul RETURN (il cursore viene portato «a capo», senza che il computer recepisca alcun comando o informazione); ma la cosa peggiore è che lo SHIFT ha effetto anche sul tasto STOP/RUN: lo STOP provoca l'interruzione del programma (break), quindi non vi sono problemi se viene premuto accidentalmente, dato che basta eseguire il comando CONT (continue) per riprendere l'esecuzione dal punto in cui è stata interrotta. La contemporanea pressione dello SHIFT, invece, nel 3032 provoca l'apparizione della scritta «press play on tape 1» e serve per il caricamento di un programma dal nastro (in caso di errore basta ripremere STOP per annullare il comando); nell'8032 lo SHIFT/RUN provoca il caricamento e l'esecuzione del primo programma memorizzato nel floppy inserito nel drive 0, quindi sostituisce la sequenza LOAD "*",8 e RUN della serie 3000: in caso di pressione accidentale, però, viene perso il contenuto della memoria, con tutti gli effetti «nefasti» che ne possono derivare (perdita definitiva del programma o dei dati non salvati). Per la verità l'utente ha qualche frazione di secondo per accorgersi dell'errore e precipitarsi a premere lo STOP, provocando il break e recuperando, con il CONT, il programma: ma bisogna essere molto veloci e... fortunati. D'accordo che premere un solo tasto per lanciare il primo programma del disco può essere di una certa comodità, ma vi rinunceremmo volentieri vista la gravità delle possibili conseguenze in caso di errore: bisognava almeno collocare il tasto in disparte, ben separato da qualsiasi altro, anziché nell'angolo superiore destro della tastiera dove è facile urtarlo premendo il DELETE o i tasti cursore. Prescindendo da questi problemi, tuttavia, non si possono non apprezzare i miglioramenti della tastiera nella serie 8000 rispetto alla serie 3000: il REPEAT e la disponibilità dei caratteri numerici anche nella tastiera principale, e non solo nel tastierino numerico, aumentano senza dubbio la praticità e la velocità di impiego. Non ci pare assolutamente un problema il fatto che, ora, sia necessario premere lo SHIFT per i caratteri che prima venivano introdotti direttamente con la prima fila di tasti (virgolette, eccetera): questa situazione è infatti del tutto normale, e comune alla maggioranza delle apparecchiature.

Rispetto al 3032, nella tastiera le diversità più evidenti sono la mancanza dei segni grafici e la presenza dei numeri anche nella fila superiore di tasti. A destra, la vista di lato mette in luce le aumentate dimensioni del monitor. L'estetica non ne risente, anzi ci sembra migliorata.

Le istruzioni print chr$(15) e print chr$(143) consentono di specificare gli angoli superiore sinistro ed inferiore destro di una «finestra» sullo schermo: l'area di scrittura viene limitata all'interno della finestra. Questa funzione serve soprattutto per aggiornare una porzione dello schermo senza cancellare la parte rimanente. Per annullare la finestra è necessario eseguire due volte il comando HOME.

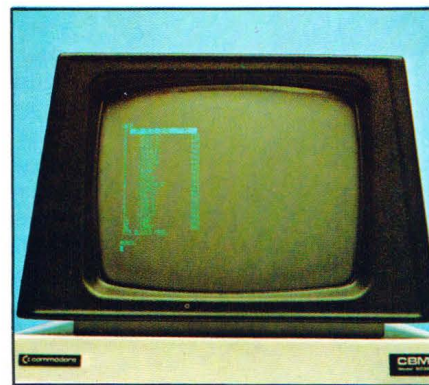
Una significativa diversità rispetto alla serie 3000 è proprio nell'effetto dello SHIFT sui tasti di caratteri alfabetici: normalmente appaiono sullo schermo le lettere minuscole, mentre lo SHIFT provoca la presentazione in maiuscolo. In pratica, come in una macchina per scrivere, ma diversamente dalla maggioranza dei computer in cui la scrittura normale è in maiuscolo e lo SHIFT serve per ottenere le minuscole. Ma nell'8032 i programmi devono essere scritti usando caratteri minuscoli (a parte ovviamente le stringhe di PRINT), contrariamente alla consuetudine; in questo modo per l'editing di programmi è necessario lo SHIFT solo per accedere ai segni posti sopra ai numeri, non per le istruzioni. Anche in questo caso avremmo preferito un funzionamento conforme a quello della maggioranza dei computer (e magari un tasto TYPEWRITER per selezionare il funzionamento tipo macchina per scrivere); tuttavia il fatto che all'accensione del sistema siano disponibili i due set maiuscolo e minuscolo, anziché maiuscolo e segni grafici come nella serie 3000, ci sembra già un buon passo avanti. I simboli grafici possono essere ottenuti eseguendo un opportuno comando di POKE, analogamente a quanto nel 3032 avviene per accedere alle minuscole.

La qualità della tastiera è rimasta invariata e, per la verità, non entusiasmante. Il principale problema è l'eccessiva rumorosità, che crea

una specie di stato di «insoddisfazione psicologica» che può far sembrare anche il «tocco» (in effetti non dei migliori) peggiore di quanto non sia realmente. Nell'uso corrente, a parte la rumorosità, non vi sono tuttavia particolari problemi di digitazione: si tratta, in sostanza, di una tastiera che si può definire abbastanza buona, nonostante l'uso non del tutto accattivante.
Veniamo, almeno brevemente, alla descrizione dell'interno della macchina. L'impostazione generale è la stessa della serie 3000, con la grossa piastra madre ed il trasformatore di alimentazione fissati alla base del contenitore di metallo. La tastiera ed il monitor sono invece fissati al coperchio, che è incerniato lungo il lato posteriore come un cofano di automobile.
Diversa, sempre con riferimento alla serie 3000, è la disposizione dei componenti sulla piastra madre: nelle foto si può vedere la fila delle RAM, le ROM (cinque più due zoccoli liberi), il microprocessore 6502 verso il fondo e, sul lato anteriore destro, il cicalino (che manca nella serie 3000).

Funzionamento e utilizzazione

All'accensione della macchina, viene emessa una segnalazione acustica e sullo schermo compare la scritta «commodore basic 4.0 - 31743 bytes free - ready». In accordo con quello che è il normale modo di visualizzazione dell'8032, i caratteri sono minuscoli. Essendo il BASIC residente su ROM, l'8032 è pronto per essere programmato. Per accedere al disco, il discorso è diverso a seconda che si possegga una «vecchia» unità (2040/3040) o una nuova (4040 e 8050). Nel primo caso è necessario aprire il canale di comando (open 1,8,15, apre il canale 1 sul dispositivo 8 con indirizzo secondario 15, che abilita l'unità a ricevere comandi) ed eseguire l'inizializzazione dei dischetti inseriti nei drive: print#1, "i" (o "i0", o "i1" per inizializzare uno solo dei due dischi). Vengono letti, nella traccia 18 del disco, il nome e l'ID (identifier) di quest'ultimo e viene caricata in memoria la mappa degli spazi disponibili (BAM, Block Availability Map), che il computer utilizza per l'accesso (in lettura o in scrittura) ai file (dati e programmi). Se si toglie il disco e se ne inserisce un altro, l'unità 2040/3040 non si accorge di questa operazione, a meno che l'ID dei due dischi non sia diverso. Questo fatto è molto importante: se l'ID corrisponde, il computer seguita a operare secondo la BAM precedentemente immagazzinata: in caso di lettura si ha semplicemente un errore, mentre in scrittura le conseguenze possono essere molto peggiori, perché le informazioni possono essere scritte sopra ad altre preesistenti, che quindi vengono perse. Per questa ragione è opportuno, quando si esegue il «format» di un disco vergine, attribuire un ID sempre diverso: se si dimentica di inizializzare il disco dopo la sostituzione in un drive, così, il sistema rivela la difformità di ID e si blocca segnalando l'errore. Con le nuove unità il problema è superato: si accorgono dell'inserimento di un disco e ne eseguono automaticamente l'inizializzazione.

A sinistra, il listing sul video di una parte del programma che genera le finestre casuali visibili nelle due foto in basso di pagina a fianco. Qui sopra, il catalog di un disco, dopo aver eseguito l'istruzione print chr$(142): i caratteri sono maiuscoli ed è stata eliminata la spaziatura fra le righe.

Non è, quindi, necessario neppure il comando print#1,"i" all'accensione della macchina. Questa procedura è non solo molto più sicura, ma anche più comoda perché semplifica l'accesso iniziale.
La nuova versione del BASIC (4.0) presenta, fra l'altro, alcuni significativi miglioramenti nella gestione del disco. Per controllare l'elenco dei file presenti su un floppy (p. es. nel drive 0), la versione 3.0 richiede i comandi LOAD"$0",8 e poi LIST: il catalog viene caricato e listato come un programma e, quindi, provoca la perdita del programma già eventualmente allocato in memoria. L'alternativa è l'uso del DOS SUPPORT, residente su un dischetto, che consente di eseguire il catalog con lo (strano) comando @$0, o @$1, o @$ (catalog di entrambi i dischi). Il nuovo BASIC accetta il comando «directory dx» (dove x è 0 o 1, il numero del drive; se omesso vengono presentati entrambi i catalog), che può essere abbreviato in «diRdx». E' possibile, indifferentemente, usare il comando «catalog» (abbreviabile in cA) al posto di directory: ma non è scritto nel manuale e, noi stessi, lo abbiamo scoperto per caso. Questa non è, a dire il vero, l'unica cosa che i manuali «nascondono»: lI, ad esempio, per il sistema significa list mentre pR vuol dire print#. Questa possibilità di abbreviazione, a quanto ci è stato detto dagli stessi responsabili della Harden, non è descritta in nessun manuale, ma viene «tramandata per via orale», ed esiste su tutti i PET, compreso il 2001.
Il DOS SUPPORT è fornito, in dotazione, anche con il BASIC 4.0; consente di utilizzare abbreviazioni per inviare alcuni comandi al disco. A parte il discorso del catalog (importante per il BASIC 3.0, ma non significativo per il 4.0), la caratteristica fondamentale consiste nel poter usare la barra e la freccia, seguite dal nome di un programma memorizzato su disco, rispettivamente per caricare e per caricare ed eseguire il programma stesso.
La nuova versione del BASIC, abbiamo detto, consente una migliore gestione della memoria di massa. Alcuni comandi sono stati aggiunti, di altri è stato semplificato l'uso. Per la formattazione e la copia dei dischi, per la cancellazione e la ridenominazione dei file, ad esempio, nella versione 3.0 la forma del comando è PRINT#1 seguito, fra virgolette, dalla stringa di controllo composta da un carattere che specifica l'operazione da eseguire (S per scratch, R per rename, N per format ecc.) e dalle indicazioni necessarie per il

completamento del comando (nome del file o dei file, drive di destinazione eccetera, a seconda della funzione richiesta). Questo discorso vale per i comandi: backup, concat, copy, rename e scratch, che ora è possibile eseguire molto più semplicemente eliminando la macchinosità del print# (tra l'altro in questa forma sono molto più facili da ricordare). Vediamoli brevemente: lo scratch serve per cancellare un file (dati o programma), il rename per cambiargli nome, il copy per duplicarlo (il nuovo file può a scelta essere creato sul medesimo disco o su quello inserito nell'altro drive); il backup serve, invece, per la copia dell'intero dischetto. Il comando concat, infine, è interessante perché consente il concatenamento di due file (deve trattarsi di file dati, di tipo sequenziale): è utile soprattutto quando si vogliono aggiungere, in coda ad un file che già esiste, delle informazioni temporaneamente immagazzinate in un «file di servizio».

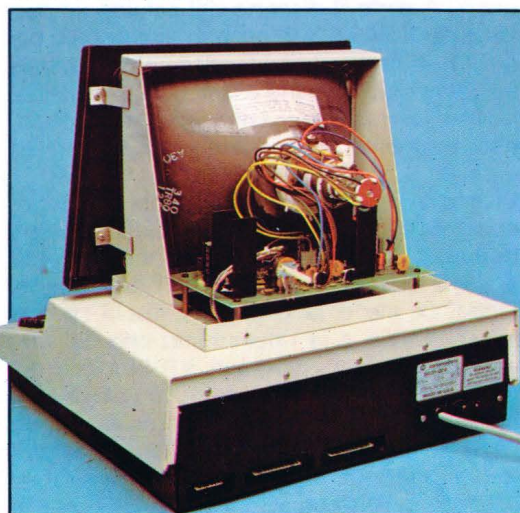
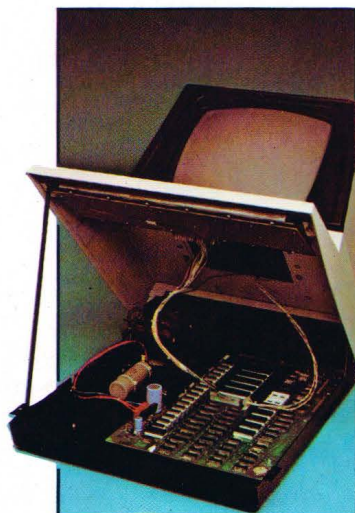
Una cosa che non ci è piaciuta: se si vuole salvare un programma con il nome di un altro già esistente, bisogna includere nel comando save o dsave il carattere «@», altrimenti il file non viene rimpiazzato. Fin qui tutto bene, anzi può essere utile dover essere consapevoli di perdere il contenuto del primo file. Ma, qualora si tenti di salvare il programma con un nome già esistente e si dimentichi di «autorizzare» la perdita di quest'ultimo, il sistema dovrebbe a nostro avviso segnalare «programma non salvato»: altrimenti, a meno di non usare il verify e se non ci si accorge in tempo dell'errore, spegnendo la macchina o richiamando un altro programma si perde definitivamente quello che si crede di aver salvato.

Sempre per il trattamento dei file sequenziali, è da segnalare che il BASIC 4.0 è stato dotato del comando append, che consente di aggiungere direttamente dei dati in coda ad un file (sequenziale). La istruzioni per la scrittura e la lettura sono le solite: print#, input# e get# (lettura un byte alla volta); il comando record#, aggiunto nel nuovo BASIC, serve per il posizionamento del puntatore per l'accesso ai file di tipo random. I classici comandi open, close, save e load sono stati affiancati dai corrispondenti dopen, dclose, dsave e dload: la differenza è che i primi possono essere rivolti anche al nastro ed è quindi necessario specificare il dispositivo cui sono riferiti (8 per il floppy), mentre i secondi servono solo per la gestione del disco e non è quindi necessario specificare l'indirizzo. Il verify è rimasto invariato (verifica che il programma in memoria sia conforme a quello registrato su disco), mentre l'header e il collect servono il primo per formattare un nuovo disco (nel 3.0 il comando corrispondente è PRINT#1,"N seguito dal nome e dall'ID da attribuire al disco), il secondo per «ripulire» il dischetto creando una nuova BAM, compattando ed eliminando gli spazi sprecati in seguito ad eventi come cattive chiusure di file o salvataggi successivi, con il medesimo nome, di programmi di diversa lunghezza. La creazione di una BAM più opportuna ha come effetto una maggiore disponibilità di spazio utile sul disco (in seguito al minore spreco) ed un aumento di velocità nel trattamento di file di una certa lunghezza, che il collect alloca in modo che vengano «spezzati» meno possibile. Il collect corrisponde, in linea di massima, al precedente comando validate.

In merito alle altre istruzioni del BASIC, non vi sono grosse diversità rispetto alla versione 3.0 ed alle possibilità degli apparecchi della stessa classe. Segnaliamo positivamente la capacità di trattare matrici fino a 255 dimensioni, in cui

Sopra: una vista della tastiera, che è fissata al coperchio del contenitore. Sotto: un particolare dell'avvisatore acustico, sullo spigolo anteriore destro del board.
Il suono emesso è una specie di cinguettio invece del classico «beep» (che avremmo giudicato preferibile).

Sopra: un primo piano della piastra madre. Notare a sinistra la fila delle RAM, a destra le cinque ROM e i due zoccoli (bianchi) liberi. Il microprocessore 6502 è verso il fondo uno dei cinque grossi integrati (quello al centro). Sotto: bastano due viti per togliere la «carrozzeria» al monitor da 12 pollici.

Qui sotto, una vista di insieme dell'8032 aperto. Il board è fissato alla base, la tastiera al coperchio del contenitore, incernierato posteriormente.

ciascun indice può, compatibilmente con la capacità di memoria, arrivare fino a 32767. Il loop (for/next/step) può avere fino a 9 livelli (loop nel loop), mentre i livelli di subroutine possono essere ben 23. Ci sembra veramente difficile mettere in crisi il CBM quanto a dimensioni di una matrice e livelli di subroutine. E' rimasto, come nelle altre versioni, l'orologio interno, accessibile per mezzo delle variabili TIME e TIME$. Non possiamo non notare, come già fatto a proposito del 3032, la mancanza dell'ON ERROR (trasferimento del programma ad una determinata linea in caso di errore), dell'AUTO (numerazione automatica delle linee in fase di stesura di un programma) e del RENUMBER (possibilità di rinumerazione delle linee del programma contenuto nella memoria centrale). Queste ultime due funzioni, per quanto riguarda il 3032, sono ottenibili per mezzo di un programma di utility in linguaggio macchina che si carica da disco, il BASIC PLUS. Supponiamo che una simile possibilità esista anche per le serie 4000 e 8000, dotate di BASIC 4.0, anche se finora non abbiamo avuto informazioni in merito. La mancanza dell'AUTO e del RENUMBER ci sembra, specie per un computer per altri versi piuttosto completo, di una certa gravità.

L'editing è molto semplice ed efficace: si lista la linea, si porta il cursore nel punto da modificare e, una volta eseguite la variazioni (usando anche, se necessario, l'insert e il delete) si preme il RETURN; è possibile, con lo stesso sistema, cambiare anche il numero della linea per duplicarla. Manca il comando per cancellare blocchi di linee (p. es. DEL 100-200 per cancellare tutte le linee dalla 100 alla 200), ma è necessario cancellarle una alla volta (100 RETURN ecc.).

Un punto di forza dell'8032, infine, è costituito dalla flessibilità e completezza della gestione del video: è possibile, ad esempio, la creazione di «finestre» di qualunque dimensione e posizione, l'inserimento e la cancellazione di una linea (con spostamento delle altre che seguono per riempire il vuoto o creare lo spazio), lo scroll (cioè il movimento di tutto il testo) verso l'alto o verso il basso. I comandi vengono impartiti portando, sotto il controllo del programma, il cursore nella posizione desiderata ed eseguendo print chr$(x), in cui x è un numero dal quale dipende la funzione da eseguire. Per annullare una finestra precedentemente creata, è necessario eseguire (o programmare) due volte di seguito il comando HOME. Certo sarebbe ancor meglio se questi comandi fossero in una forma tale da poterli ricordare più facilmente, ma ci si può senza dubbio accontentare. Anzi, l'8032 è dotato di una gestione del video veramente di alto livello, di certo fra le migliori per un personal computer. L'istruzione (diretta o da programma) print chr$(142) consente l'accesso ai segni grafici, che divengono richiamabili con lo shift mentre il set delle maiuscole si sostituisce a quello delle minuscole: coesistono, dunque, maiuscole e segni grafici, ma non minuscole e segni grafici. Alcuni di questi, peraltro, non sono accessibili direttamente da tastiera ma solo come chr$ (quelli che nella serie 3000 appaiono come shift del tastierino numerico); questo non ci sembra affatto un problema vista la comodità del fatto che lo shift non agisca sul numeric pad. Il print chr$(14) ripristina il funzionamento normale. Il set grafico può essere selezionato anche per mezzo di un opportuno POKE; la differenza sta nel fatto che il chr$(142) ha anche l'effetto di eliminare lo spazio fra una riga e l'altra, consentendo la realizzazione di disegni «pieni» (come nella foto di apertura, ad esempio). E' stato aggiunto, rispetto alla serie 3000, un cicalino accessibile con il chr$(7): il suono, simile ad una specie di cinguettio, non ci piace anche perché di durata a nostro avviso eccessiva (circa mezzo secondo). Avremmo preferito il classico «beep» che si può sempre ripetere più volte se si ha bisogno di una segnalazione di maggior lunghezza. Durante la scrittura, il cicalino si attiva automaticamente quando mancano cinque spazi alla fine di una linea; una funzione che, a dire il vero, non riteniamo di fondamentale utilità, tanto più che il cursore si arresta mentre viene emesso il cinguettio, quindi se si scrive con una certa rapidità può accadere di saltare delle lettere. Una cosa che sicuramente si apprezza, nell'uso, è la velocità del sistema e, particolarmente, della presentazione sul video.

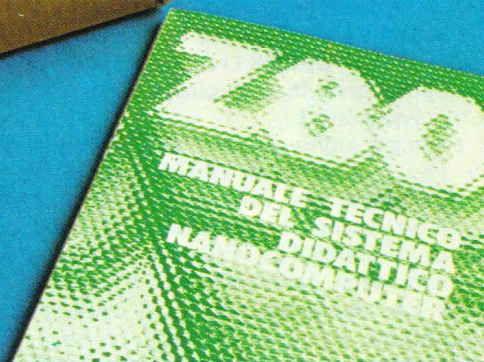
Conclusioni

La serie 8000 rappresenta indubbiamente un significativo passo in avanti rispetto alla serie 3000. Il video da duemila caratteri, la tastiera ora finalmente standard «davvero», le possibilità del nuovo BASIC 4.0 e la flessibilità della gestione del video hanno potenziato le capacità del sistema aumentandone anche la praticità di uso. Il discorso vale non solo per l'unità centrale 8032, ma anche per il doppio driver 8050 (con DOS 2.5), capace di tenere in linea oltre un milione di caratteri, che fra le altre prestazioni esegue automaticamente (finalmente...) l'inizializzazione del disco introdotto.

Ci sono ovviamente dei difetti, dei quali abbiamo parlato nel testo: l'impressione che abbiamo, valutandoli nell'insieme, è che alla Commodore quando si studiano le macchine e i sistemi operativi si dica, ad un certo punto, basta senza aver voglia, dopo aver «fatto trenta», di «fare trentuno». E così, una macchina priva di grossi difetti di base si ritrova una serie di piccole pecche che in fase di studio sarebbe stato facile eliminare. Mettere su ROM un renumber, una segnalazione di file già esistente sarebbe costato pochissimo, ma avrebbe tolto al sistema limitazioni nelle quali l'utente può imbattersi con una certa frequenza.

Questo discorso non cambia la positività del giudizio sulla macchina, anche considerando il prezzo che ci pare più che ragionevole: due milioni e trecentomila lire per l'8032, altrettanto, grosso modo, per l'8050. Un megabyte e ottanta colonne per circa cinque milioni ci sembrano un discorso decisamente molto interessante.

Marco Marinacci

SISTEMA DIDATTICO

SGS NANOCOMPUTER

Costruttore: SGS — ATES Componenti Elettronici S.p.A. — Via C. Olivetti, 2 — 20041 Agrate (Milano)

Prezzi:

NBZ80-S — Nanocomputer S — L. 825.000
NBZ80 — scheda base + miniterminale — L. 471.000
NBZ80-A — NBZ80 + alimentatore — L. 566.000
NBZ80-B — NBZ80 + contenitore/alimentatore — L. 636.000
UPZ80-BS — upgrading da B ad S — L. 306.000
UPZ80-S — upgrading da NBZ80 a NBZ80-S — L. 412.000
NEZ80 — breadboard — L. 236.000
RCZ80 — registratore a cassette — L. 84.000
WIOZ80 — cavo per registratore — L. 23.000
KNZ80 — kit da NBZ80 a CLZ80 4/2 — L. 295.000
CLZ80-4 — scheda base completa con 4K RAM — L. 528.000
RAZ80-48 — memoria da 48K RAM — L. 1.325.000
PIZ80 — scheda di I/O — L. 550.000
VDZ80 — scheda video con tastiera — L. 550.000
FLZ80 — controllo floppy disk — L. 380.000
CPZ80 — Cestello γ bus per 4 schede — L. 186.000

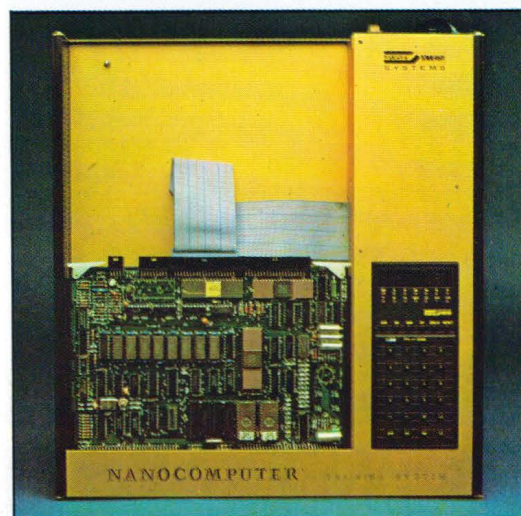
Riferimento servizio lettori: 33

Tutti i maggiori produttori di microprocessori hanno in catalogo delle schede microcomputer. Come mai dei costruttori di semiconduttori sentono il bisogno di costruire anche schede? Il motivo è essenzialmente uno solo: la fortuna di un microprocessore, misurata in numero di pezzi venduti, è legata più o meno direttamente al numero di progettisti hardware e software in grado di implementare sistemi intorno ad esso.
Un fattore eminentemente culturale che condiziona pesantemente dei risultati industriali. Ecco quindi nascere gli «Evaluation board», i «Single board computer» e i sistemi didattici, tre categorie di schede microcomputer che spesso si differenziano più per particolari marginali che non nell'hardware di base del sistema.
Le schede di valutazione si rivolgono, almeno nelle intenzioni, a progettisti che desiderano acquisire dimestichezza con una famiglia di microprocessori, i single board computer a costruttori di sistemi che per il ridotto numero di unità prodotte preferiscono concentrare i propri sforzi nello sviluppo del software più che in quello dell'hardware.
Il sistema didattico è in pratica un evaluation board nel quale, presupponendo di rivolgersi ad utenti inizialmente digiuni della materia, vengono semplificati al massimo i problemi hardware (p.e. corredando la scheda con l'alimentatore) e curata particolarmente la letteratura.
Dopo aver scelto di puntare sul microprocessore Z 80, forse il più noto e popolare (una bella lotta con il 6502 e il 6800!) dei microprocessori a 8 bit, di fronte al problema «culturale» dei progettisti, la SGS ha evidentemente deciso di partire con un'operazione di largo respiro puntando su quello che è forse il più didattico dei sistemi didattici oggi sul mercato: il nanocomputer.

Probabilmente, per la maggior parte di noi, il Nanocomputer è un sistema a microprocessore composto da un contenitore metallico nel quale sono montate, protette da un coperchio di perspex, due schede a circuiti stampati: una contenente il microprocessore e la circuiteria relativa, l'altra il breadboard per gli esperimenti hardware, con una serie di interuttori, pulsanti e indicatori luminosi. In un vano del contenitore è installato il miniterminale esadecimale, e al suo interno l'alimentatore.
Questa immagine è relativa al Nanocomputer S, ove S sta per super, la configurazione più completa, e più costosa.
Il Nanocomputer può comunque essere aquistato in altre tre diverse configurazioni:
NBZ80 comprendente solo piastra base e miniterminale;
NBZ80-A come l'NBZ80, ma con l'alimentatore;
NBZ80-B come l'NBZ80, ma inserito nel contenitore/alimentatore.
Le configurazioni più ridotte possono evolversi

in quelle maggiori mediante l'acquisto di kit di trasformazione ed è quindi possibile distribuire nel tempo la spesa.
Per completezza, fisseremo la nostra attenzione sul Nanocomputer S, che, essendo la configurazione più alta, comprende tutte le parti delle configurazioni minori.
Una precisazione prima di proseguire: il Nanocomputer è un sistema didattico: questo non vuol dire che sia un oggetto primitivo o malfatto: la piastra su cui si basa è la stessa del sistema microcomputer CLZ80, nata per l'uso professionale, e verso cui il sistema può evolversi.
Il Nanocomputer realizza in questo senso quello che dovrebbe essere lo scopo di ogni buon maestro: adeguare le proprie capacità apparenti a quelle dell'allievo, essendo sempre un po' (e soltanto un po') più bravo di lui.
Il Nanocomputer S. Con il Nanocomputer S, oltre ai componenti fisici della macchina (scheda base, terminale esadecimale, breadboard, contenitore/alimentatore) viene fornita una ricca documentazione comprendente il «manuale tecnico del sistema didattico Nanocomputer» (una meticolosa descrizione dell'hardware del sistema), il Nanobook 1 e 3 e il catalogo delle parti disponibili per arricchire il sistema.
Il Nanobook 1, con le sue 256 pagine è una dettagliatissima guida alla programmazione della CPU Z80, attraverso più di 100 esperimenti, scritto in modo accurato (la raccomandazione di non far cadere in terra la macchina è annoverata tra le cose che non debbono essere fatte) e piacevole, ed in lingua italiana, è certamente uno dei motivi determinanti nella scelta del Nanocomputer. Le medesime caratteristiche presenta il Nanobook 3, una guida all'interfacciamento dello Z80. Il Nanobook 2 offre una panoramica delle tecniche digitali necessarie al buon uso di un microprocessore. Allegato al primo volume troviamo un pieghevole con tabelle relative alle istruzioni dello Z80, per un rapido passaggio da linguaggio assembler a codice macchina e viceversa.
Per mettere in funzione la macchina è sufficiente connettere la presa di corrente, e azionare l'interruttore con spia incorporata posto sul retro del contenitore (se possediamo una configurazione più debole del Nanocomputer S dovremo connettere il miniterminale nell'apposito connettore ed allacciarci ad un alimentatore a 4 tensioni). Basterà poi schiacciare il tasto di RESET per far partire il programma di monitor residente sulla scheda.
La scheda viene fornita con 4K di RAM e 2K di EPROM. La RAM è dedicata per intero a contenere i programmi utente, eccetto le locazioni dello stack ed alcune locazioni di servizio per il monitor.
Naturalmente i 2K di EPROM contengono il monitor, che provvede alla gestione del miniterminale attraverso una PIO (parallel input output: vale a dire un circuito integrato che mette a disposizione 16 linee di ingresso/uscita, di cui la scheda monta due esemplari), e all'implementazione di un'UART (universal asynchronous receiver transmitter: un dispositivo che permette la trasmissione e la ricezione di dati seriali) software, sempre attraverso PIO, per il comando del registratore a cassette.
Quest'ultimo, venduto sotto la sigla RCZ80, è un registratore portatile Philips opportunamente modificato (allineamento testine, squadratore di ingresso, teleaccensione del motore) dalla SGS. Il prezzo, tenuto conto del costo degli adattamenti, può essere considerato «politico» (in questo caso basso).

Il miniterminale ed il programma di monitor consentono all'utente il colloquio con la macchina. Attraverso questi l'operatore può

Il Nanocomputer S, la versione più completa del Nanocomputer. In alto la scheda con il microprocessore e le memorie, in basso la scheda con il «bread board» con connessioni a pressione per gli esperimenti, a destra il miniterminale: sul pannello di perspex che protegge le schede sono serigrafate le indicazioni per il riconoscimento dei vari componenti.

Il Nanocomputer B. In pratica la versione S senza scheda per esperimenti (e senza pannello di plastica). Con il kit di espansione UPZ80-S è possibile trasformare in qualsiasi momento il Nanocomputer B in Nanocomputer S.

NBZ-80: scheda e miniterminale. Un Nanocomputer ridotto all'osso per il quale occorre acquistare (o costruire) un alimentatore. Esiste anche un Nanocomputer A composto da scheda, terminale e alimentatore.

caricare il proprio programma in RAM, nell'area da lui scelta, visualizzare i registri, ed eventualmente modificarne il contenuto, fissare i break-points e far avanzare il programma passo passo. L'aspetto del miniterminale è quello di un calcolatore da tasca, con due file di sette led ciascuna sotto il display ad otto digit, e con una tastiera a 30 tasti. All'interno contiene soltanto degli array di transistor per il pilotaggio degli indicatori luminosi, e un circuito di decodifica che completa la logica di multiplaggio (è una logica abbastanza complessa, dato che consente la gestione di ben 98 elementi, tra tasti e diodi luminescenti, con l'uso di sole 12 linee).

Il display può essere idealmente diviso in due gruppi di indicatori di 4 digit ciascuno. Al gruppo di sinistra è affidato il compito di rappresentare gli indirizzi a 16 bit in notazione esadecimale, mentre il gruppo di destra visualizza il contenuto delle locazioni di memoria, dei registri e delle porte di I/O. I led, eccettuata l'indicazione di errore (unico led verde) indicano la natura della grandezza visualizzata dal display. Se, per esempio, è acceso il led AF, nella parte destra del display sarà visualizzato il contenuto dei due registri A ed F. Se è acceso il led MEM, il contenuto del display rappresenta un indirizzo di memoria (a sinistra) e il contenuto della locazione stessa (a destra).

La confezione del Nanocomputer è particolarmente curata. Tutto il sistema, manuali compresi, viene fornito in una bellissima valigetta di «cartone ondulato di plastica» con tanto di manico. Sembra un particolare trascurabile, ma invece non lo è: spesso riporre quando non la si usa una scheda a microprocessore comporta il pericolo di danneggiarla o il disagio di reimballarla.

Il miniterminale del Nanocomputer è probabilmente la più rifinita tra le tastiere esadecimali oggi sul mercato: è costruito come le calcolatrici «di una volta» con contattiera a pressione dorata e senza risparmio nei materiali. Smontandola si ha l'impressione che robustezza e affidabilità siano del massimo livello. Il miniterminale è il primo strumento di colloquio con la macchina. I led IR AF BC DE HL IX IY SP PC indicano che sul display è visualizzato il contenuto di un registro. ARS segnala che il registro visualizzato fa parte del set di registri alternativo. BRK indica che il terminalino è nel modo «breakpoint». MEM ed I/O segnalano la presenza sul display di un contenuto di memoria o di una porta di I/O. Il led ERR si accende quando viene compiuta un'operazione illecita.

I tasti possono essere divisi in cinque classi:

16 tasti esadecimali, con i quali si introducono le grandezze numeriche.

5 tasti per il controllo del visualizzatore e dell'input da tastiera: →, ←, ARS selezionano la grandezza visualizzata o da scrivere, 2ND permette l'ingresso del 2° byte nei registri a 2 bytes; BRK consente l'introduzione dei breakpoints.

4 tasti per il controllo del programma: SS (esecuzione della singola istruzione), GO (via al programma), BREAK (stop al programma) e RESET.

3 tasti per l'introduzione dei dati: LA, carica un indirizzo; ST, memorizza un dato in memoria, in un registro o in una porta di I/O e incrementa l'indirizzo; INC, incrementa l'indirizzo visualizzato.

2 tasti per il controllo dell'interfaccia seriale, DP e LD (dump e load), permettono l'uso del registratore a cassette.

1 interuttore TTY-CASS permette la selezione tra un terminale seriale (TTY) e un registratore audio a cassette (CASS).

L'uso della macchina dal terminale è abbastanza agevole; si sente forse la mancanza di un tasto di decremento dell'indirizzo, da contrapporre al tasto INC, per consentire una rapida esplorazione della memoria nei due sensi, che, stranamente, non è stato previsto.

Per quanto riguarda il programma di monitor che gestisce il miniterminale bisogna notare che il single step ed i breakpoints non volatili (che, cioè si conservano anche dopo che l'istruzione puntata è stata superata), sono strumenti di potenza non indifferente nella messa a punto dei programmi. Ciò avviene a scapito della completa utilizzabilità, da parte dell'utente, delle risorse della CPU (ad esempio l'istruzione FF assume il significato di «ritorno al monitor», invece di essere un salto incondizionato, mentre alcune locazioni di memoria non possono essere utilizzate). Queste limitazioni rappresentano comunque una perdita esigua paragonate alle possibilità offerte in cambio.

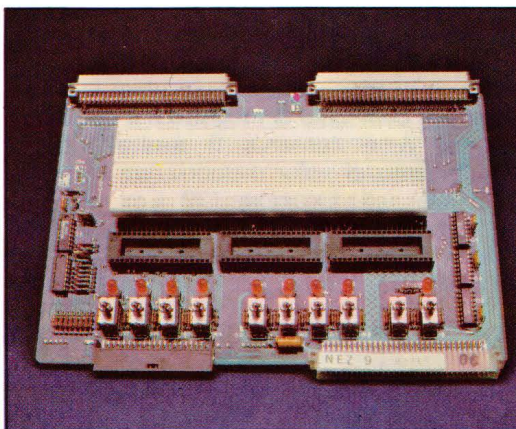
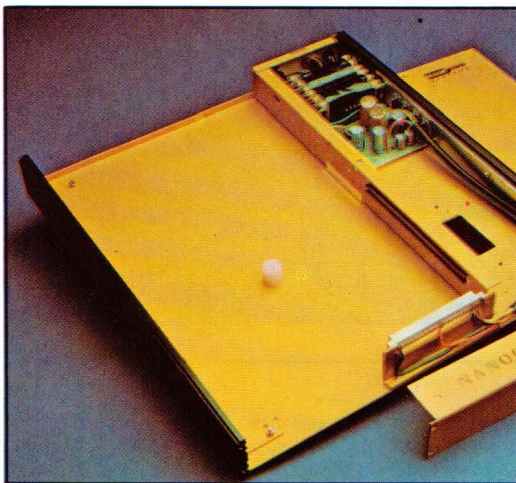
La scheda madre. È utile cercare di guardare a fondo la costituzione fisica del Nanocomputer, perché conoscere un calcolatore non vuol dire soltanto conoscerne il linguaggio, e un sistema didattico deve fornire la conoscenza completa di una CPU e del modo di progettare con essa, e non semplicemente insegnare a programmare.

Ecco un rapido elenco delle caratteristiche più salienti della scheda madre:

CPU	Z80
interfaccia con l'esterno	2 PIO, un'UART (da montare) interfaccia per registratore a cassette
RAM	fino a 16K di RAM dinamica
ROM	fino ad 8K di EPROM

La piastra è letteralmente irta di connettori di ottima qualità, per un totale di ben 234 linee. Su queste linee compaiono tutti i segnali da e verso la CPU (dopo aver transitato attraverso drivers TTL), i segnali delle porte e dell'interfaccia seriale.
L'insieme è abbastanza classico, con certe facilitazioni che ne rendono l'uso agevole per chi deve disporre rapidamente di una scheda a microprocessore. La piastra prevede, infatti, un insieme di ponticelli, da collegarsi con una goccia di stagno, che consentono le seguenti operazioni:
— montaggio di EPROM diverse come capacità ed alimentazione (2708, 1K, triplice alimentazione; 2716, 2K, singola o triplice alimentazione);
— scelta dell'indirizzo da cui far partire l'area di EPROM (di 4K in 4K nel caso si montino EPROM da 1K; di 8K in 8K nel caso si montino EPROM da 2K);
— possibilità di montare RAM dinamiche da 4 o 16K (4027 o 4116);
— scelta dell'indirizzo da cui far partire l'area di RAM (di 4K in 4K o di 16K in 16K);
— scelta del tipo di interfaccia seriale e del baud rate (nel caso si completi la scheda montando l'UART).
A questo bisogna aggiungere la possibilità di cambiare l'entry point sostituendo l'integrato Q 48, una piccola PROM bipolare 6331, che contiene l'indirizzo a cui il program counter andrà a posizionarsi dopo il reset. In essa sono immagazzinate, a partire dall'indirizzo 0, le parole in esadecimale C3 02 FC, che corrispondono ad un salto alla locazione FC02; tutte le altre locazioni sono vuote e, comunque, non vengono mai viste dalla CPU. Per modificare l'entry point dobbiamo scrivere, o far scrivere, in una 6331 vergine le parole:
C3
xx parte meno significativa dell'entry point
yy parte più significativa dell'entry point
Questa flessibilità porta la piastra al di là del semplice sistema didattico, e la rende utilissima nel caso si disponga di software per lo Z80, e si debba imbastirgli intorno un hardware adeguato.

La scheda per esperimenti è collocata nella posizione più comoda per l'operatore e si collega alla scheda madre direttamente (elettricamente e meccanicamente) attraverso i connettori J1 e J2 che assumono lo stesso nome e portano le stesse linee (γbus) sia sulla scheda madre che sulla scheda per esperimenti.
Attraverso un cavo piatto giungono alla scheda per esperimenti i segnali provenienti dalla PIO dedicata all'utente; un led indica la presenza dell'alimentazione a + 5 V.
Una basetta per montaggi sperimentali senza saldature è posta al centro della scheda. Su questa lo sperimentatore può inserire direttamente i componenti relativi all'esperienza, traendo i segnali e le alimentazioni da tre zoccoli a 40 pin sui quali confluiscono tutti i segnali che arrivano alla scheda. Il circuito sperimentale potrà essere controllato attraverso 8 interruttori e 2 pulsanti ciascuno collegato ad un circuito di antirimbalzo a flip flop set reset. I segnali presenti sul circuito verranno rivelati da una serie di 8 led, ciascuno pilotato da un driver TTL.
È da notare la presenza, sul circuito stampato, nei pressi di J1 e J2, del posto riservato per due connettori, che, una volta montati, vengono ad essere in parallelo a J1 e J2 e consentono l'inserimento di una terza scheda con lo stesso standard.

La scheda contenente la CPU è il cuore del sistema. Appositi ponticelli consentono di impiegare tutti i principali tipi di EPROM, PROM, ROM e RAM dinamiche. I due connettori in basso costituiscono il γ BUS che consente di collegare alla CPU altre schede del sistema.

Sulla scheda esperimenti sono riportati su zoccolo tutte le linee del γ BUS e della PIO a disposizione dell'utente. Interruttori e pulsanti con Flip Flop antirimbalzo, led pilotati da buffer, il bread board a pressione permettono di realizzare con minima perdita di tempo un notevole numero di esperimenti.

Il pieghevole contenente l'elenco delle 158 istruzioni dello Z-80.

Un esperimento col nanocomputer. Supponiamo di essere entrati in possesso del Nanocomputer e di voler verificare che la CPU Z80, di cui abbiamo tanto sentito parlare, ma che non abbiamo mai usato, sia in grado, per esempio, di eseguire un'addizione.

Per far questo accendiamo l'alimentatore. L'apparecchio non dà segni di vita a parte l'accensione di un led sulla scheda Esperimenti. Questo vuol dire che la CPU non sta eseguendo il programma monitor, che, tra l'altro, gestisce il display.

Basta premere il tasto RESET: il display si accende e la macchina è pronta a funzionare. Naturalmente sul terminale non esistono tasti operativi e, se vogliamo eseguire un'addizione, dobbiamo memorizzare l'istruzione corrispondente. Sul pieghevole allegato al Nanobook scopriamo l'esistenza dell'istruzione ADD A,B (con codice macchina 80) che somma il contenuto del registro B a quello del registro A e conserva il risultato in A. Supponiamo di voler memorizzare la nostra istruzione nella locazione 1ØØ Con i tasti →, ←, facciamo in modo che si accenda il led MEM: questo indica che il visualizzatore rappresenta una locazione di memoria. Per far sì che la locazione visualizzata sia la 1ØØ, introduciamo il valore Ø1ØØ, con la tastiera esadecimale, e premiamo il tasto LA (load address). A questo punto il display indicherà a sinistra il valore Ø1ØØ (indirizzo), mentre a destra leggeremo il contenuto della locazione. Ora possiamo introdurre il codice della nostra istruzione, 8Ø, attraverso la tastierina, e lo memorizziamo nella locazione puntata attraverso il tasto ST. Introduciamo il valore FF e memorizziamolo con ST. Questo dice al monitor di riprendere il controllo una volta che la nostra istruzione è stata eseguita. Ora dobbiamo caricare i contenuti dei registri A e B. Supponiamo di mettere 3 in A e 4 in B. La sequenza dei tasti è la seguente:

led su AF (usando i tasti →, ←)
3
2ND indica che si vuole scrivere il byte più significativo di un registro a 16 bit
ST
led su BC
4
2ND
ST

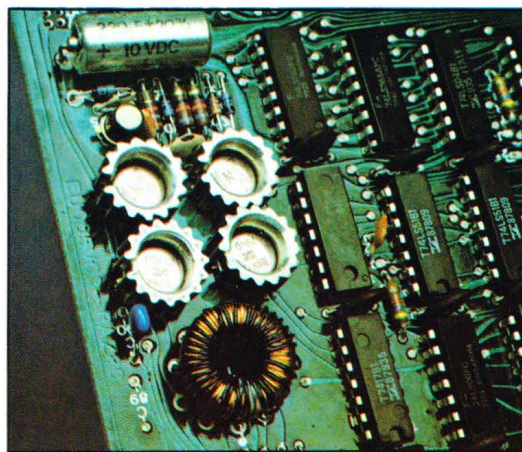
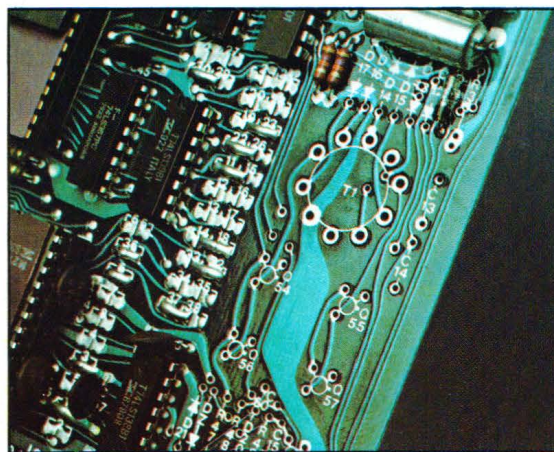
A questo punto possiamo partire con l'esecuzione della nostra istruzione:

led su PC si comanda la visualizzazione del program counter
Ø1ØØ si memorizza nel program counter
ST l'indirizzo di partenza
GO si comanda l'esecuzione.

Se tutte le operazioni sono state eseguite correttamente, posizionando il led su AF, dovremo vedere, nel registro A il risultato della operazione 3+4=7.

Tutte queste operazioni possono sembrare molto macchinose, ma un piccolo periodo di tirocinio le rende completamente automatiche. Non dobbiamo comunque dimenticare che lo scopo principe dell'MBZ80 è l'insegnamento delle caratteristiche della CPU Z80, per il quale un immediato impatto con un sistema operativo evoluto sarebbe probabilmente controproducente dal punto di vista didattico.

Possibilità di espansione. Premesso che sarebbe un errore accostarsi al Nanocomputer esclusivamente con l'idea di costruirvi intorno un «personal», questo non vuol dire che non sia possibile farlo; anzi, una volta raggiunta l'indispensabile conoscenza di base, grazie al

La scheda NBZ-80 può essere trasformata nella CLZ-80 (scheda unità centrale del sistema industriale SGS) integrandola con l'alimentatore «on board» per alimentazione singola a 5 volt e sostituendo la EPROM del monitor. Il kit di conversione si chiama KNZ-80.

CPU Z80

Lo ZØ0 è probabilmente il microprocessore a 8 bit più noto al momento, e forse quello sul quale è stata spesa più carta: può sembrare superfluo parlarne ancora. Tuttavia la scelta di chi si avvicina ad un sistema didattico è motivata, oltre che dalla generica bontà del sistema, dal tipo di CPU che questo gli permetterà di conoscere. Usare l'NBZ80 vuol dire usare la CPU Z80: infatti il programma di monitor consente di comunicare con la CPU, ma non ne arricchisce la potenza, che rimane quella della CPU «nuda».

Un motivo importante per la scelta di questo microprocessore è la compatibilità software con l'8080, per il quale esiste una vasta mole di programmi già pronti. I programmi scritti per l'8080 girano felicemente sullo Z80, che possiede rispetto al primo un set di istruzioni quasi doppio. Chi abbia acquistato un'esperienza sull'8080, ad esempio attraverso la lettura dei Bugbooks, dovrebbe trovare nello Z80 la naturale espansione della sua conoscenza.

Chi guarda soprattutto la parte hardware potrà essere attratto dal fatto che dalla CPU escono tutte le linee necessarie al suo utilizzo, senza la necessità di strani multiplexaggi e dai contatori, interni al chip, che consentono una facile rifresco delle memorie dinamiche. Molto superficialmente, ecco le caratteristiche dello Z80:

lunghezza della parola 8 bit
memoria indirizzabile 64 Kbytes
registri per un totale di 208 bit
interruzioni 3 diversi modi
istruzioni 150
alimentazione singola a 5V

Tutto questo dà un'idea molto pallida delle possibilità del componente: forse è più utile una considerazione in generale: per chi si accosta ai microprocessori, lo Z80 rappresenta un validissimo compromesso tra potenza e semplicità sia per la parte hardware sia, specialmente, per la programmazione. *G.R.*

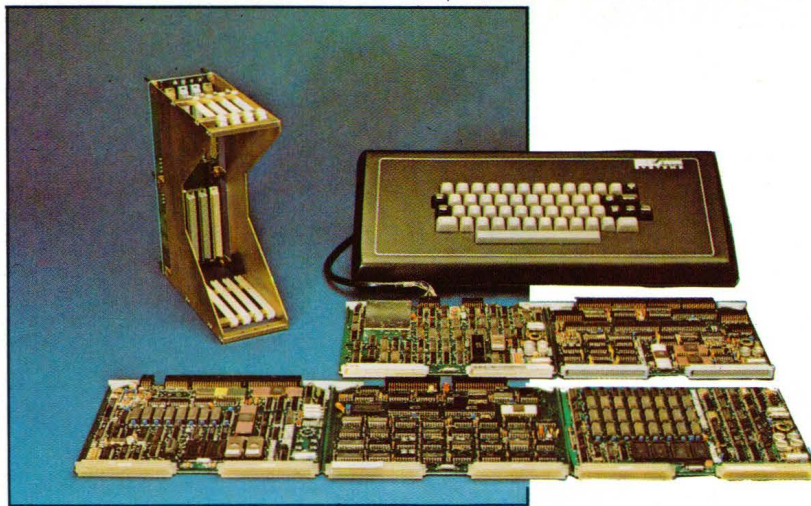
Alcune schede del sistema industriale SGS (memoria da 48 K, unità centrale, interfaccia video con tastiera, controller per floppy disc, I/O. A sinistra, 4 schede inserite nel cestello CPZ-80.

tirocinio con il Nano, potremo tranquillamente far crescere il sistema, sicuri che la conoscenza precedentemente acquisita ci permetterà di comprendere appieno le risorse che la nuova configurazione ci mette a disposizione.

La piastra che siamo abituati a chiamare Nanocomputer (a meno di una manciata di componenti che possono facilmente essere montati negli spazi appositi) è la scheda microcalcolatore CLZ80, base del sistema che porta lo stesso nome. Infatti l'inserimento del miniterminale (caratteristica del Nanocomputer) non modifica in alcun modo la circuiteria della piastra: tutto ciò che è necessario alla gestione del terminale è contenuto nelle EPROM del monitor: una volta che queste siano state rimosse la piastra non conserverà più memoria del suo precedente stato di «Nanocomputer» (a differenza di quanto avviene per molte altre schede microcomputer con tastiera e display su scheda).

Un primo passo può essere l'uso del kit di conversione KNZ8Ø, che fornisce i componenti mancanti per completare la piastra (avremo finalmente la singola alimentazione e l'UART) e 2K di monitor su EPROM. A questo punto dovremo mettere nel cassetto il miniterminale e procurarci un terminale vero. Il collegamento con un terminale seriale non è assolutamente problematico: il nostro calcolatore parla TTY, RS232, e fornisce livelli TTL con un baud-rate da 5Ø a 96ØØ. Uniformandosi alla linea di prodotto SGS ATES si potrà optare per l'unità VDZ8Ø, basata sempre sulla CPU Z8Ø, con 64 caratteri per 16 linee, che contiene già un modulatore video sulla banda VHF.

Una crescita ulteriore può essere rappresentata dalla scheda controllo dei floppy disk, FLZ80, che può gestire fino a quattro drivers, per una capacità complessiva di 2.5 Mbytes. Per usare i floppy e caricare, eventualmente, il sistema operativo in RAM, è indispensabile disporre di espansioni di memoria. Allo scopo esiste una cartolina, la RAZ80, che può ospitare fino a 48K di RAM dinamica e 8K di EPROM.

Se le PIO contenute nella scheda base fossero insufficienti per le nostre applicazioni (automazione, controlli di processo), potremo arricchire il sistema con il modulo PIZ80, che contiene 4 PIO, interfaccia seriale (SIO) e timer.

Tutto il sistema può trovare un comodo alloggio nei cestelli offerti allo scopo che possono ospitare 4 od 8 schede (rispettivamente CPZ80 e CTZ80).

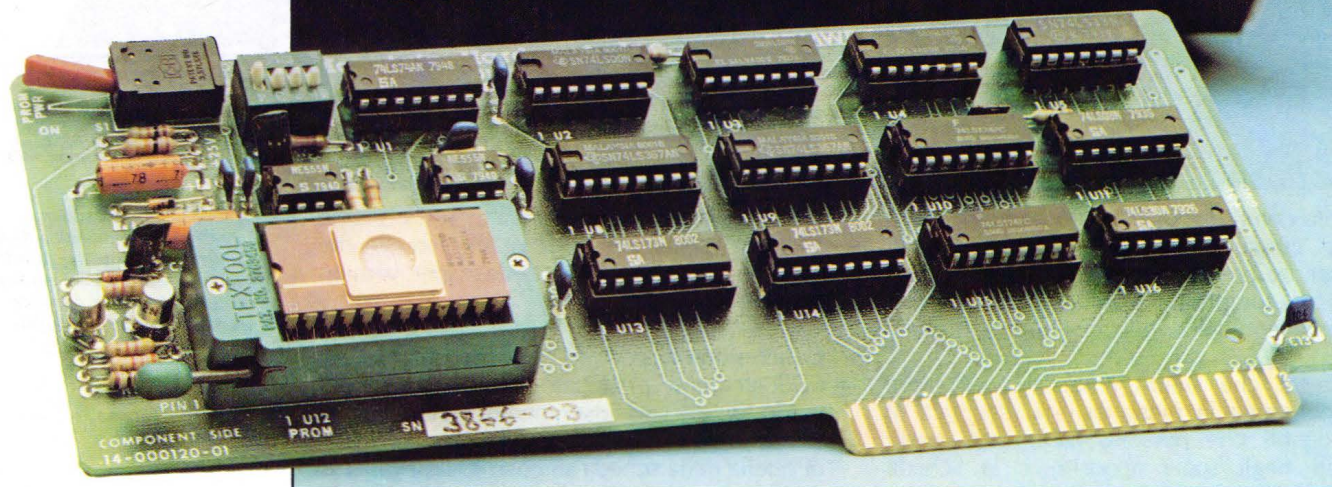
Conclusioni. Il prodotto Nanocomputer è stato evidentemente progettato con l'obiettivo primario di un suo inserimento nella struttura scolastica. La robustezza del telaio maggiore di quella della scheda singola, la possibilità di eseguire esperimenti senza dover prendere in mano il saldatore, la progressione didattica dei manuali, parlano chiaro. Ma queste stesse caratteristiche, unitamente alle eccellenti doti di flessibilità della scheda NBZ80, ne fanno un prodotto degno del massimo interesse anche per chi, al di fuori di una struttura scolastica, vuole imparare ad usare lo Z80 e la famiglia di componenti ad esso associati. La possibilità di reimpiegare la scheda base, altre applicazioni come un già disponibile gioco degli scacchi su ROM, l'impiego come minisistema di sviluppo e le espansioni verso linguaggi ad alto livello (BASIC già disponibile, Assembler in arrivo), lo rendono meno soggetto ad una obsolescenza del tipo «ormai ho imparato, non mi serve più».

Né bisogna trascurare il vantaggio di imparare a conoscere come le proprie tasche non solo lo Z80, ma anche una scheda che è il fulcro di un sistema per applicazioni industriali tra i più completi e rifiniti.

Nel confrontare il prezzo del Nanocomputer S con quello di altre schede microcomputer occorre non trascurare le differenze tra un evaluation board e un vero sistema didattico, differenze che pur marginali dal punto di vista sistemistico, incidono pesantemente dal punto di vista costruttivo e quindi sui costi. Un paragone più diretto può essere quello con la scheda NBZ80 completa di terminale, ma anche in questo caso occorre tenere conto che il terminale del Nanocomputer ha un valore notevolmente superiore alle tastiere esadecimali normalmente montate sulle schede di valutazione. In quest'ottica, tenuto conto della qualità e quantità dei materiali impiegati, il prezzo del Nanocomputer, in prima istanza più elevato di quello di altri sistemi (peraltro non direttamente confrontabili), appare pienamente giustificato.

Gianni Romeo

UN EPROM PROGRAMMER PER APPLE

Mountain Hardware ROMWRITER

Il ROMWRITER è una scheda che può essere inserita in uno degli slot dell'Apple e consente di trasferire su EPROM dei programmi in linguaggio macchina. Può essere utile per avere su ROM, anziché su disco, utility di uso frequente: il renumber, ad esempio.

Costruttore:
Mountain Hardware
300 Harvey
West Boulevard
Santa Cruz, CA 95060

Distrubutore:
Softec
Corso M. D'Azeglio, 60
10126 Torino

Prezzo:
L. 384.000 + IVA

Riferimento servizio lettori: 34

La Mountain Hardware è una ditta specializzata nella progettazione e la produzione di periferiche per microcomputer e piccoli computer in genere. Tra le varie periferiche importate in Italia c'è il Supertalker (un'interfaccia che permette all'Apple di «parlare»), l'Appleclock (un orologio quarzato in tempo reale con calendario fino ad un anno), la ROM Plus (una scheda speciale con decodifica per contenere fino a 12 K di ROM) oltre naturalmente al programmatore di EPROM in prova questo mese.

Descrizione e uso

L'EPROM programmer è costituito da un'unica scheda da inserire in uno degli slot dell'Apple. In basso a sinistra si trova uno zoccolo «Zero Insertion Force», cioè a forza di inserzione zero che permette da inserire e sfilare una EPROM migliaia di volte, senza che né lo zoccolo né l'integrato vengano danneggiati; inoltre ha i contatti dorati per garantire la massima affidabilità. In alto a sinistra si trova un deviatore che serve per togliere l'alimentazione della EPROM durante l'inserimento o l'asportazione mentre il computer è acceso. La tensione di 25 V necessaria per la programmazione della EPROM è generata sulla scheda da un convertitore cc/cc alimentato dalla tensione di 12 V presente sul BUS dell'Apple. Accanto al deviatore di alimentazione si trova un microdeviatore a quattro vie di cui, però, solo due sono collegate. Una serve come Write Protect, cioè per proteggere la EPROM da eventuali riprogrammazioni accidentali (ed è utile abituarsi a togliere la protezione solo durante la programmazione); l'altra serve per l'abilitazione/disabilitazione dell'indirizzo CFFF (torneremo su questo argomento più avanti). Infine si trovano 13 integrati digitali (TTL Scottky) usati in pratica per realizzare una porta bidirezionale indirizzabile a 8 bit oltre alla decodifica del BUS indirizzi per selezionare la EPROM. Notiamo quindi che questa scheda non ha una ROM da 256 byte come è consueto nelle interfacce per l'Apple, normalmente chiamate interfacce intelligenti. Infatti il software necessario per il funzionamento del programmatore di EPROM è fornito su un disco (senza trucchi, cioè facilmente copiabile per fare il Backup).

Il programma, chiamato PROMBURNER, è facilissimo da usare. Supponiamo di aver inserito la scheda nello slot 3 (può essere usato qualsiasi slot da 1 a 6), ed inserita una EPROM, con il computer ancora spento oppure con il deviatore di alimentazione della EPROM in posizione off. Inseriamo ora il disco Eprom Writer ed accendiamo il computer. Scriviamo RUN PROMBURNER e vediamo apparire una alla volta una serie di domande che chiedono il numero dello slot, l'indirizzo di inizio di programmazione della EPROM, l'indirizzo di fine della EPROM, se si vuole caricare da disco i dati da trasferire su EPROM oppure se trasferire un'area di memoria. Nell'ultimo caso viene richiesto l'indirizzo di partenza dell'area di RAM. Infine l'Apple chiede conferma che tutto sia giusto prima di passare alla programmazione della EPROM. Sullo schermo appare:

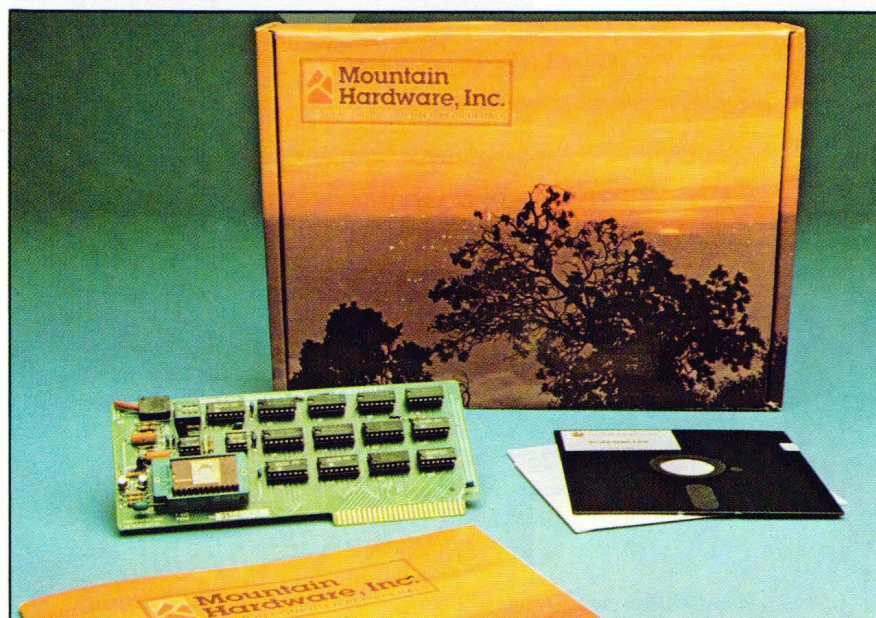
ENTER EPROM PROGRAMMER SLOT # (1-7)?
ENTER EPROM STARTING ADDR. IN HEX?
ENTER EPROM ENDING ADDR. IN HEX?
DO YOU WANT TO LOAD A PROGRAM (Y OR N)?
ENTER PROG. STARTING ADDR. IN HEX?
ARE THE ADDRESSES CORRECT (Y OR N)?

Come si può vedere è possibile scegliere tra programmare tutta la EPROM (C800-CFFF) oppure solo una parte, specificandone gli indiriz-

zi di partenza e di fine che, comunque, devono essere compresi tra C000 e CFFF. A volte è comodo copiare su EPROM più di un programma o serie di dati ed è quindi di fondamentale importanza che sia possibile programmare anche solo una parte della EPROM. L'area di memoria da copiare nella EPROM deve naturalmente contenere il programma che deve essere trasferito. Se già lo contiene, per esempio se è stato assemblato da un assembler oppure inserito a mano, basta dare l'indirizzo di partenza di quest'area di memoria e dopo aver dato conferma che tutto è esatto avrà inizio la programmazione della EPROM. I dati da copiare possono anche essere caricati da un file binario, basta rispondere «Y» alla domanda «DO YOU WANT TO LOAD A PROGRAM (Y OR N)?», ed inserire il nome del file. Dopo aver dato la conferma che tutti gli indirizzi sono esatti avrà inizio la programmazione: il tempo necessario è di circa 2 minuti per una intera EPROM. Durante questo periodo il cursore lampeggiante sparisce dallo schermo; avremmo preferito per esempio un contatore di indirizzo o qualche altra indicazione dinamica per indicare che il computer sta lavorando e che non sia «inchiodato». Ricordiamo che l'Eprom Writer è stato progettato per programmare solo le EPROM del tipo 2716 da 2K. Durante la programmazione possono verificarsi due situazioni. Se il deviatore di Write Protect è rimasto nella posizione ON oppure se si è dimenticato di «accendere» la EPROM con il deviatore dell'alimentazione apparirà il seguente messaggio:
BAD BURN OF EPROM!
DO YOU WANT TO TRY AGAIN (Y OR N)?
Cioè programmazione della EPROM mal riuscita!; vuoi riprovare?
Quindi prima di rispondere Y bisogna essere certi che i due deviatori siano nella posizione giusta. L'altro problema che si può verificare è che gli indirizzi selezionati non siano stati cancellati se, per esempio, è stata inserita una EPROM sbagliata oppure se la EPROM è stata esposta alla luce ultravioletta per troppo poco tempo, e quindi non può essere programmata. Sullo schermo apparirà il seguente messaggio:
EPROM ADDRESSES SELECTED AREN'T ERASED!
SAVE WHOLE EPROM WITH CHANGES (Y - N)?
Cioè le locazioni di memoria selezionate per la programmazione non sono state cancellate. Vuoi salvare su disco la EPROM con le modifiche?
In altre parole abbiamo la possibilità di salvare su disco un «imagine» della EPROM con il contenuto della EPROM così come era più i dati da inserire nelle locazioni selezionate. Possiamo poi cancellare la EPROM e riprogrammarla con i dati salvati precedentemente su disco. Questo metodo è utile anche per modificare alcune locazioni di memoria di una EPROM già programmata. Supponiamo di voler cambiare le locazioni da C900 a C90F, (cioè 16 locazioni) di una EPROM già programmata. Inseriamo la EPROM e facciamo girare il programma specificando, per la gamma di indirizzi da programmare, da C900 a C90F. Il programma ovviamente si ferma indicando che le locazioni di memoria selezionate non sono cancellate. Salviamo quindi tutta la EPROM con i cambiamenti su un file sul disco, poi cancelliamo la EPROM. Dopo averla reinserita bisogna far ripartire il programma PROMBURNER, perché sarà necessario specificare gli indirizzi di partenza e di fine come C000 e CFFF; carichiamo quindi il file precedentemente salvato e dopo la programmazione abbiamo la nostra EPROM modificata.

A che cosa serve una EPROM?

Nel numero 4 di micro & personal COMPUTER abbiamo descritto che cos'è una EPROM ed anche un metodo economico per cancellarla; era anche evidente a che cosa potrebbe servire essendo compatibile con le ROM già esistenti sull'AIM 65. Ma con un EPROM programmer collegato all'Apple, le quali ROM non sono più compatibili con la EPROM 2716, che cosa si può fare? Data la enorme facilità di uso del ROMWRITER si può usarlo per programmare delle EPROM per uso in altri sistemi; oppure si possono sviluppare dei programmi, metterli su EPROM ed inserirli nella scheda ROMPLUS (sempre della Mountain Hardware) che ospita un massimo di 6 EPROM. Si potrebbe quindi per esempio scrivere un nuovo sistema operativo e metterlo su EPROM... Una caratteristica fondamentale del ROMWRITER è che può essere usato anche per testare una EPROM appena programmata: questo è molto utile quando, per esempio, si sta sviluppando una interfaccia intelligente che impiegherà magari una ROM da soli 256 byte. Come esempio di uso della scheda proviamo a trasferire su EPROM il programma di RENUMBER, cioè di rinumerazione delle righe di programma in Applesoft, in maniera che al momento dell'accensione della macchina abbiamo subito a disposizione il Renumber senza doverlo prima caricare dal disco e senza che ci «mangi» 2K byte di memoria RAM. A questo punto bisogna precisare che è necessario usare il programma di Renumber che sta sui dischi DOS 3.2 e non quello sui dischi più recenti chiamati DOS 3.2.1, che è più lungo (il funzionamento è lo stesso) e non entra nei 2 K di memoria della EPROM.

Foto 1. L'EPROM programmer viene fornito con un eccellente manuale e di un disco contenente il software necessario per il funzionamento.

```
|LIST
20  FOR I = 36062 TO 36311
30  IF  PEEK (I) = 142 AND ( PEEK
    (I - 1) = 181 OR  PEEK (I -
    1) = 182 OR  PEEK (I - 1) =
    183 OR  PEEK (I - 1) = 180) THEN
     PRINT  PEEK (I): POKE I,2
40  NEXT
50 I = 34875
60 A =  PEEK (I): IF A = 76 OR A =
    32 OR A = 221 THEN 90
70 I = I + 1: IF I = 36581 THEN 1
    40
80  GOTO 60
90 B =  PEEK (I + 2): IF B > 135 AND
    B < 145 THEN  PRINT  PEEK (I
     + 1): GOTO 110
100  GOTO 70
110  IF  PEEK (I + 1) + 59 < 255 THEN
     POKE I + 2,B + 64: POKE I +
    1, PEEK (I + 1) + 59: GOTO 7
    0
120  IF  PEEK (I + 1) + 59 > 255 THEN
     POKE I + 2,B + 65: POKE I +
    1, PEEK (I + 1) + 59 - 256
130  GOTO 70
140  POKE 36567,240: POKE 36568,2
    06
```

List 1. Programma in Applesoft per la rilocazione del programma in linguaggio macchina.

```
 8800.883A
8800- A9 4C 8D F5 03 A9 35 8D
8808- F6 03 20 4A FF 78 20 58
8810- FF BA BD 00 01 29 0F 09
8818- C0 8D F7 03 A9 F0 85 36
8820- A9 FD 85 37 20 EA 03 A9
8828- 8D 20 F0 FD A9 00 8D B4
8830- 02 8D B5 02 60 8D FF CF
8838- 4C 3B C8
*
```

List 2. Codice oggetto da inserire a partire dalla locazione 8800 con i comandi del monitor.

List 3. Listato disassemblato del codice oggetto di fig. 2.

```
 8800LL
8800-   A9 4C       LDA   #$4C
8802-   8D F5 03    STA   $03F5
8805-   A9 35       LDA   #$35
8807-   8D F6 03    STA   $03F6
880A-   20 4A FF    JSR   $FF4A
880D-   78          SEI
880E-   20 58 FF    JSR   $FF58
8811-   BA          TSX
8812-   BD 00 01    LDA   $0100,X
8815-   29 0F       AND   #$0F
8817-   09 C0       ORA   #$C0
8819-   8D F7 03    STA   $03F7
881C-   A9 F0       LDA   #$F0
881E-   85 36       STA   $36
8820-   A9 FD       LDA   #$FD
8822-   85 37       STA   $37
8824-   20 EA 03    JSR   $03EA
8827-   A9 8D       LDA   #$8D
8829-   20 F0 FD    JSR   $FDF0
882C-   A9 00       LDA   #$00
882E-   8D B4 02    STA   $02B4
8831-   8D B5 02    STA   $02B5
8834-   60          RTS
8835-   8D FF CF    STA   $CFFF
8838-   4C 3B C8    JMP   $C83B
```

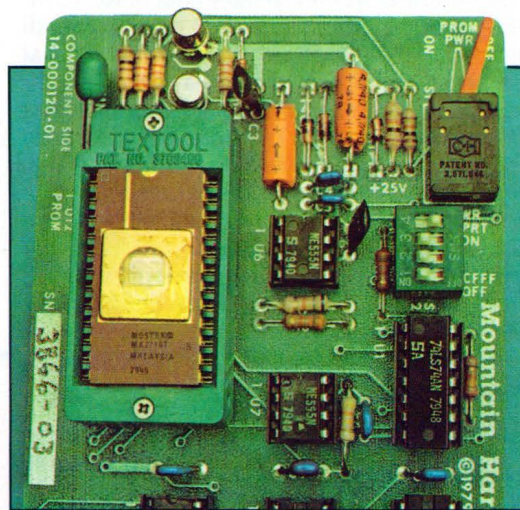



Foto 2. Particolare del deviatore a quattro posizioni e del deviatore di accensione della EPROM. Inoltre si può notare lo zoccolo di buona qualità con i contatti dorati che garantiscono la massima affidabilità.

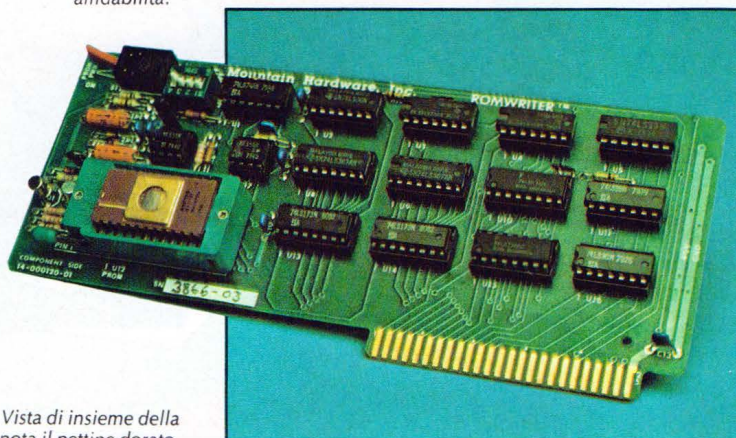



Foto 3. Vista di insieme della scheda; si nota il pettine dorato.

La mappa di memoria dell'Apple

Prima di procedere vediamo come è strutturata la mappa di memoria dell'Apple, ed in particolare quella relativa alle periferiche. Ricordiamo che una pagina di memoria è un gruppo di 256 locazioni consecutive. In pagina zero, cioè da $0000 a $00FF, troviamo delle variabili fondamentali usate dal monitor (sistema operativo di base) dal DOS e dal linguaggio in uso in un dato momento (per esempio Applesoft). Tra queste variabili ci sono due locazioni di memoria ($36 e $37) che definiscono un indirizzo a 16 bit. Questo indirizzo si chiama CWS ed è quello relativo all'output dei caratteri. Se l'output è diretto allo schermo, questo indirizzo è $FDF0. Quando invece l'uscita viene diretta verso una periferica, come per esempio una stampante, questo indirizzo può essere cambiato in $CN00, dove N è il numero dello slot dell'interfaccia. Il cambiamento avviene automaticamente quando dal BASIC si effettua l'istruzione PR#N (dove N è sempre il numero dello slot). Pagina uno è usata dal microprocessore per lo stack e, come tale, è riservata. Pagina due è l'input buffer. Ogni volta che viene premuto un tasto, il codice ASCII di quel tasto viene immagazzinato a partire da $200 fino ad un massimo di $2FF. La terza pagina viene usata durante il Booting del DOS. Nell'ultima parte di questa pagina si trovano dei puntatori importanti: in particolare $3D0 contiene l'indirizzo di una routine che riconnette il DOS se per qualche motivo viene perso (dal monitor prémi 3D0G), $3EA serve per trasferire il contenuto di CWS e KSW (l'input switch) ai registri usati dal DOS, ed infine $3F5 che contiene l'indirizzo della routine chiamata con l'istruzione «&» dell'Applesoft. Le successive quattro pagine sono dedicate al video (24 per 40 caratteri). A $800, cioè a partire dalla pagina 8, comincia l'area di RAM utente, normalmente usata per programmi in BASIC e variabili, display grafico etc. In una macchina da 48K la RAM arriva fino a $BFFF. Altri 4K della mappa di memoria sono dedicati alle periferiche. Per ogni slot N, c'è una pagina compresa tra $CN00 e $CNFF per N da 0 a 7. Ogni interfaccia può avere una ROM da 256 byte contenente il software necessario per fare funzionare l'interfaccia, dal che deriva il termine Interfaccia Intelligente. Se non dovessero bastare i 256 byte di memoria, si può includere una ROM da 2K mappata da $C800 a $CFFF. Poiché tutte le interfacce possono avere una ROM da 2K è necessario che venga adottato un metodo di indirizzamento particolare per evitare conflitti tra una ROM e l'altra. Il metodo è il seguente: su ogni interfaccia che comprende una ROM da 2K, ci deve essere un Flip-Flop settato quando l'interfaccia viene indirizzata nella gamma di indirizzi base per quella interfaccia. Supponiamo di avere una interfaccia nello slot 2. Quando viene indirizzata una locazione di memoria compresa tra $C200 e $C2FF il Flip-Flop viene settato: questo a sua volta abilita la ROM da 2K, che una volta abilitata (tramite il Flip-Flop) rimane abilitata finché non si resetta il Flip-Flop. Inoltre, su ciascuna interfaccia ci deve essere un circuito che riconosce l'indirizzo $CFFF: questo segnale viene usato per disabilitare (o

```
]CALL-151
*8800.8FFF
8800- A9 4C 8D F5 03 A9 35 8D
8808- F6 03 20 4A FF 78 20 58
8810- FF BA BD 00 01 29 0F 09
8818- C0 8D F7 03 A9 F0 85 36
8820- A9 FD 85 37 20 EA 03 A9
8828- 8D 20 F0 FD A9 00 8D B4
8830- 02 8D B5 02 60 8D FF CF
8838- 4C 3B C8 A0 00 84 C1 20
8840- B7 00 C9 4D D0 03 4C 55
8848- CD A0 01 B1 67 D0 04 88
8850- 4C D2 CC A9 FF A2 0A A0
8858- 00 85 9D 85 9E 84 9F 84
8860- A0 86 A1 84 A2 86 A3 84
8868- A4 20 B7 00 90 29 F0 57
8870- C9 48 D0 03 4C DE CC C9
8878- 43 D0 0D 4C D5 CD 20 B7
8880- 00 C9 2C D0 12 20 B1 00
8888- A2 03 DD E5 CE F0 0D CA
8890- 10 F8 E8 C9 80 F0 05 A0
8898- CF 4C D2 CC 8A 0A A8 20
88A0- B1 00 F0 06 90 0A C9 2C
88A8- D0 F5 A2 00 86 A8 F0 05
88B0- 20 04 CE A6 A7 96 9D A6
88B8- A8 96 9E E0 FA 90 05 A0
88C0- D6 4C D2 CC AA D0 B7 A5
88C8- A1 05 A2 D0 05 A0 EC 4C
88D0- D2 CC A2 03 BD 67 00 9D
88D8- A5 00 CA 10 F7 86 AB A0
88E0- 02 B1 A5 C8 C5 9F B1 A5
88E8- C8 E5 A0 B0 47 20 49 CE
88F0- D0 ED A0 15 4C D2 CC A2
88F8- 00 86 AB A0 02 B1 A5 81
8900- A7 C8 20 58 CE C0 03 F0
8908- F4 A5 A3 81 A7 20 58 CE
8910- A5 A4 81 A7 20 58 CE 20
8918- 49 CE F0 2E 18 A5 A3 65
8920- A1 85 A3 A5 A4 65 A2 85
8928- A4 B0 04 C9 FA 90 05 A0
8930- 27 4C D2 CC A5 9D A0 02
8938- D1 A5 C8 A5 9E F1 A5 B0
8940- B6 A5 AB 10 05 A0 15 4C
8948- D2 CC A2 00 A9 FF 81 A7
8950- 20 58 CE A9 FF 81 A7 20
8958- 58 CE A5 67 85 A5 A5 68
8960- 85 A6 A0 02 84 AB B1 A5
8968- 85 9D C8 B1 A5 85 9E 20
8970- AB CE B0 0E A0 00 84 AB
8978- 20 AB CE 90 05 A0 3B 4C
8980- D2 CC 20 49 CE D0 DB A5
8988- A7 85 A5 A5 A8 85 A6 A5
8990- 67 85 B8 A5 68 85 B9 A5
8998- 69 85 A3 A5 6A 85 A4 D0
89A0- 0C C0 EF 90 05 A0 DE 4C
89A8- D2 CC 20 B1 00 20 B1 00
89B0- A8 D0 03 4C 50 CA 20 B1
89B8- 00 20 B1 00 A0 04 20 B1
89C0- 00 C8 F0 E1 AA F0 DA 10
89C8- F5 A2 06 DD E9 CE F0 05
89D0- CA 10 F8 30 E9 20 B1 00
89D8- B0 E7 20 04 CE A5 A7 85
89E0- 9D A5 A8 85 9E 20 70 CE
89E8- A2 00 86 A9 86 AA 86 AB
89F0- A2 0F 06 9D 26 9E F8 A5
89F8- A9 65 A9 85 A9 A5 AA 65
8A00- AA 85 AA A5 AB 65 AB 85
8A08- AB D8 CA 10 E5 E8 86 9D
8A10- A2 02 A9 01 85 9E BD A9
8A18- 00 4A 4A 4A 4A 29 0F C5
8A20- 9D F0 08 C6 9D C8 D0 03
8A28- 4C A5 C9 BD A9 00 C6 9E
8A30- F0 EB CA 10 DD A5 9D D0
8A38- 03 C8 F0 EC 20 B7 00 C9
8A40- C9 F0 07 C9 2C F0 03 4C
8A48- C1 C9 C8 F0 DB 4C D5 C9
8A50- A2 FF 18 B5 74 95 9E F5
8A58- A6 95 A2 E8 F0 F5 B0 05
8A60- A0 52 4C D2 CC E9 02 B0
8A68- 0F A0 96 20 D6 CE 20 0C
8A70- FD C9 72 F0 03 4C BB CC
8A78- A5 A5 85 9F A5 A6 85 A0
8A80- A0 00 C6 9E C6 A0 88 B1
8A88- 9F 91 9D 98 18 65 9F 45
8A90- 67 D0 08 A5 A0 69 00 45
8A98- 68 F0 05 98 D0 E8 F0 E2
8AA0- 38 A5 69 65 A1 85 A3 A5
8AA8- 6A 65 A2 85 A4 A5 67 85
8AB0- A5 65 A1 85 B8 A5 68 85
8AB8- A6 65 A2 85 B9 90 14 18
8AC0- 98 A0 00 65 A5 91 A5 C8
8AC8- AA A5 A6 69 00 91 A5 86
8AD0- A5 85 A6 20 B1 00 20 B1
8AD8- 00 A8 D0 03 4C 8B CB 20
8AE0- B1 00 85 9D 20 B1 00 85
8AE8- 9E 20 70 CE A0 02 A5 9D
8AF0- 91 A5 C8 A5 9E 91 A5 C8
8AF8- 20 B1 00 91 A5 C8 AA F0
8B00- BE 10 F5 A2 06 DD E9 CE
8B08- F0 05 CA 10 F8 30 E9 20
8B10- B1 00 B0 E7 20 04 CE A5
8B18- A7 85 9D A5 A8 85 9E 20
8B20- 70 CE A2 00 86 A9 86 AA
8B28- 86 AB A2 0F 06 9D 26 9E
8B30- F8 A5 A9 65 A9 85 A9 A5
8B38- AA 65 AA 85 AA A5 AB 65
8B40- AB 85 AB D8 CA 10 E5 E8
8B48- 86 9D A2 02 A9 01 85 9E
8B50- BD A9 00 4A 4A 4A 4A 29
8B58- 0F C5 9D F0 07 C6 9D 09
8B60- 30 91 A5 C8 BD A9 00 C6
8B68- 9E F0 EC CA 10 DE A5 9D
8B70- D0 05 A9 30 91 A5 C8 20
8B78- B7 00 C9 C9 F0 07 C9 2C
8B80- F0 03 4C FB CA 91 A5 C8
8B88- 4C 0F CB 18 A5 A5 69 02
8B90- 85 69 85 AF A5 A6 69 00
8B98- 85 6A 85 B0 A9 00 A0 07
8BA0- 91 A5 88 10 FB 38 A5 73
8BA8- E9 06 85 9D A8 A5 74 E9
8BB0- 00 85 9E C4 69 E5 6A E9
8BB8- 01 B0 05 A0 52 4C D2 CC
8BC0- A0 05 A9 00 91 9D 88 10
8BC8- FB A5 67 85 9F 85 A5 A5
8BD0- 68 85 A0 85 A6 A0 00 84
8BD8- A1 84 A2 C8 20 55 CE D0
8BE0- 03 4C BB CC A0 02 A5 A1
8BE8- D1 A5 C8 A5 A2 F1 A5 B0
8BF0- 11 B1 A5 85 A2 88 B1 A5
8BF8- 85 A1 A5 A5 85 9F A5 A6
8C00- 85 A0 20 49 CE D0 DD A0
8C08- 00 38 B1 9F 85 A5 E5 9F
8C10- 85 A3 C8 B1 9F 85 A6 E5
8C18- A0 85 A4 A5 9D 85 A7 E5
8C20- A3 85 9D A5 9E 85 A8 E5
8C28- A4 85 9E A4 A3 88 B1 9F
8C30- 91 9D 88 D0 F9 B1 9F 91
8C38- 9D A5 A5 85 A7 A5 A6 85
8C40- A8 20 49 CE A0 00 B1 A7
8C48- 91 9F C8 AA D0 F8 C0 05
8C50- 90 F4 A5 A5 05 A6 F0 16
8C58- 18 98 A0 00 65 9F 91 9F
8C60- AA C8 A9 00 65 A0 91 9F
8C68- 86 9F 85 A0 D0 CB A0 01
8C70- B1 67 F0 03 4C C9 CB A5
8C78- 67 85 A5 A5 68 85 A6 38
8C80- A5 9D E9 01 85 B8 A5 9E
8C88- E9 00 85 B9 A0 00 20 B1
8C90- 00 91 A5 C8 AA D0 F7 C0
8C98- 05 90 F3 84 AB A0 01 20
8CA0- 55 CE F0 17 18 A0 00 A5
8CA8- AB 65 A5 91 A5 AA C8 A9
8CB0- 00 65 A6 91 A5 86 A5 85
8CB8- A6 D0 D1 A9 EF 85 C1 A2
8CC0- 0E A9 00 95 9D 9D 00 02
8CC8- CA 10 F8 85 B8 A2 02 86
8CD0- B9 60 20 2D FF 20 48 F9
8CD8- 20 D6 CE 4C BB CC AD B4
8CE0- 02 0D B5 02 F0 05 A0 85
8CE8- 4C D2 CC A5 73 C5 AF A5
8CF0- 74 E5 B0 E9 04 B0 05 A0
8CF8- 52 4C D2 CC A5 AF E5 67
8D00- 8D B4 02 A5 B0 E5 68 8D
8D08- B5 02 A5 73 8D B6 02 A5
8D10- 74 8D B7 02 A0 00 A5 73
8D18- D0 02 C6 74 C6 73 A5 AF
8D20- D0 02 C6 B0 C6 AF B1 AF
8D28- 91 73 A5 67 C5 AF A5 68
8D30- E5 B0 90 E2 A5 67 69 02
8D38- 85 AF 85 69 A5 68 69 00
8D40- 85 B0 85 6A A0 01 A9 00
8D48- 91 67 88 10 FB A0 60 20
8D50- D6 CE 4C BB CC AD B4 02
8D58- 0D B5 02 F0 F5 38 A4 73
8D60- A5 74 E9 01 C4 69 E5 6A
8D68- B0 05 A0 52 4C D2 CC A5
8D70- 67 A6 68 85 A1 86 A2 A0
8D78- 01 B1 A1 F0 07 AA 88 B1
8D80- A1 4C 73 CD A5 A1 85 42
8D88- A5 A2 85 43 AD B6 02 85
8D90- 3E AD B7 02 85 3F A5 73
8D98- 85 3C A5 74 85 3D A0 00
8DA0- 20 2C FE AD B6 02 85 73
8DA8- AD B7 02 85 74 38 AD B4
8DB0- 02 E9 02 85 A1 AD B5 02
8DB8- E9 00 85 A2 18 A5 AF 65
8DC0- A1 85 AF 85 69 A5 B0 65
8DC8- A2 85 B0 85 6A A9 00 8D
8DD0- B4 02 8D B5 02 A5 67 85
8DD8- A1 A5 68 85 A2 18 A0 01
8DE0- B1 A1 F0 1D A0 04 C8 B1
8DE8- A1 D0 FB C8 98 65 A1 AA
8DF0- A0 00 91 A1 A5 A2 69 00
8DF8- C8 91 A1 86 A1 85 A2 90
8E00- DD 4C 9C CB A2 00 86 A7
8E08- 86 A8 AA 06 A7 26 A8 B0
8E10- 31 A5 A8 85 AB A5 A7 0A
8E18- 26 AB B0 26 0A 26 AB B0
8E20- 21 65 A7 85 A7 A5 AB 65
8E28- A8 85 A8 B0 15 8A 29 0F
8E30- 65 A7 85 A7 A5 A8 69 00
8E38- 85 A8 B0 06 20 B1 00 90
8E40- C9 60 68 68 A0 D6 4C D2
8E48- CC A0 00 B1 A5 C8 AA B1
8E50- A5 86 A5 85 A6 B1 A5 60
8E58- E6 A7 D0 02 E6 A8 A5 A7
8E60- C5 73 A5 A8 E5 74 90 07
8E68- 68 68 A0 52 4C D2 CC 60
8E70- 84 AB A5 A3 85 A7 A5 A4
8E78- 85 A8 A0 00 B1 A7 45 9D
8E80- D0 07 C8 B1 A7 45 9E F0
8E88- 15 18 A5 A7 69 04 85 A7
8E90- 90 02 E6 A8 A0 01 B1 A7
8E98- C9 FF D0 DE F0 0A C8 B1
8EA0- A7 85 9D C8 B1 A7 85 9E
8EA8- A4 AB 60 A5 69 85 A3 A5
8EB0- 6A 85 A4 A4 AB 18 B1 A3
8EB8- 45 9D D0 07 C8 B1 A3 45
8EC0- 9E F0 12 A5 A3 69 04 85
8EC8- A3 90 02 E6 A4 A0 01 B1
8ED0- A3 C9 FF D0 DE 60 B9 F0
8ED8- CE 48 C8 09 80 20 ED FD
8EE0- 68 10 F3 60 00 45 53 49
8EE8- 46 85 AB AC B0 BC C4 00
8EF0- 4E 4F 20 50 52 4F 47 52
8EF8- 41 4D 20 49 4E 20 4D 45
8F00- 4D 4F 52 59 8D 4E 4F 20
8F08- 4C 49 4E 45 53 20 49 4E
8F10- 20 52 41 4E 47 45 8D 49
8F18- 4E 43 52 45 4D 45 4E 54
8F20- 20 54 4F 4F 20 4C 41 52
8F28- 47 45 8D 44 55 50 4C 49
8F30- 43 41 54 45 20 4C 49 4E
8F38- 45 20 4E 55 4D 42 45 52
8F40- 53 8D 4F 55 54 20 4F 46
8F48- 20 4D 45 4D 4F 52 59 8D
8F50- 50 52 4F 47 52 41 4D 20
8F58- 4F 4E 20 48 4F 4C 44 2C
8F60- 20 55 53 45 20 22 26 4D
8F68- 22 20 54 4F 20 52 45 43
8F70- 4F 56 45 52 8D 48 4F 4C
8F78- 44 20 46 49 4C 45 20 49
8F80- 4E 20 55 53 45 8D 4C 49
8F88- 4D 49 54 45 44 20 4D 45
8F90- 4D 4F 52 59 2C 20 4D 41
8F98- 59 20 44 45 53 54 52 4F
8FA0- 59 20 50 52 4F 47 52 41
8FA8- 4D 20 20 20 20 20 43 4F
8FB0- 4E 54 49 4E 55 45 20 28
8FB8- 59 2F 4E 29 3F 07 8D 53
8FC0- 59 4E 54 41 58 8D 3E 20
8FC8- 36 33 39 39 39 8D 4C 49
8FD0- 4E 45 20 54 4F 4F 20 4C
8FD8- 4F 4E 47 8D 49 4E 43 52
8FE0- 45 4D 45 4E 54 20 3D 20
8FE8- 30 8D FF FF 20 50 52 4F
8FF0- 47 52 41 4D 20 20 20 20
8FF8- 20 43 4F 4E 54 49 4E 55
*
```

Codice oggetto del programma di Renumber da memorizzare nella EPROM.

resettare) il Flip-Flop. La prima cosa da fare all'inizio del programma contenuto nella ROM è di riferirsi all'indirizzo $CFFF (ad esempio con una istruzione LDA $CFFF o STA $CFFF) per disabilitare tutte le ROM. Questo naturalmente disabilita anche la ROM stessa ma al successivo riferimento all'indirizzo base (in questo caso tra $C200 e $C2FF) viene riabilitata e si può così godere di 2K di memoria mappata da $C800 fino a $CFFE.

A questo punto possiamo, finalmente, capire a cosa serve il deviatore CFFF-OFF presente sulla scheda del ROMWRITER: quando è nella posizione ON è abilitato il Flip-Flop, questa è quindi la posizione da usare quando si usa la scheda come porta ROM. Quando invece è necessario programmare anche l'ultimo byte (CFFF) allora si deve mettere il deviatore nella posizione CFFF-OFF.

Il Renumber dell'Apple

Il programma di RENUMBER incluso sul disco DOS 3.2 è un programma «strano». Si carica come un programma Applesoft ma, quando viene listato, è molto corto e contiene apparentemente solo degli statement di PRINT. Il segreto è che alla fine c'è un programma rilocabile in linguaggio macchina, che viene locato negli ultimi due K di RAM immediatamente sotto HIMEM, adattandosi quindi ad un sistema di qualsiasi grandezza. Un Apple con 48K ha un HIMEM di $9600 e il programma viene trasferito tra $8E00 e $9600. Una volta trasferito il programma non è più rilocabile, vale a dire che se lo spostassimo a $C800 - $CFFF non funzionerebbe più a causa dei salti con indirizzamento assoluto, e quindi sarà necessario modificare quelle locazioni di memoria che contengono riferimenti assoluti. Inoltre è necessario aggiungere un programmino, sempre in linguaggio macchina, in testa al programma in maniera da inizializzare i vettori (cioè indirizzi) di CWS quando viene eseguita l'istruzione PR#N, dove come al solito N è il numero dello slot. Il programma di inizializzazione è lungo 59 byte, quindi bisogna rilocare il programma Renumber non a $C800 ma a $C83B. Purtroppo facendo questo il programma non entra più nei due K della EPROM, ma poiché ci sono circa 200 byte liberi nella penultima pagina possiamo comprimerlo come vedremo in seguito.

Il trasferimento su EPROM

Vediamo ora in pratica come possiamo realizzare tutto questo. Per non dover cambiare tutte le locazioni assolute a mano abbiamo scritto un piccolo programma di rilocazione in Applesoft (vedi figura 1). Prima di cominciare conviene inserirlo in memoria e salvarlo su disco con il nome, per esempio, RILOC. Inoltre bisogna introdurre il programma di inizializzazione riportato in figura 2. Per fare ciò scriviamo CALL-151 dall'Applesoft, per entrare in monitor (identificato dall'asterisco). Il programma (figura 3) va inserito partendo da $8800, quindi digitiamo:
*8800: A9 4C 8D F5 03...etc...4C 3B C8 (Return)
Dopo aver inserito tutti i dati possiamo confrontarli con il listato, premendo 8800L. Quando siamo sicuri che tutto è corretto digitiamo 3D0G per tornare in Applesoft. Salviamo su disco scrivendo:
BSAVESTART. CODE, A$8800, L $3B (Return)
Il programma di Renumber deve essere rilocato a partire da $C83B, ma poiché non abbiamo RAM a quella locazione lo allochiamo a $883B. Questo si ottiene facilmente, specificando HIMEM a $903B, scrivendo:
HIMEM:36923 (Return)
Inseriamo il disco contenente il programma di Renumber e scriviamo:
RUNRENUMBER (Return)
Poi cambiamo le locazioni assolute con:
RUNRELOC (Return)
Vediamo apparire una serie di numeri sullo schermo e quando ritorna il cursore dell'Applesoft scriviamo:
BLOADSTART.CODE (Return)
per aggiungere in cima il programmino di inizializzazione. Pazienza, abbiamo quasi fatto... ci manca ancora di spostare l'area di memoria da $8F3B - $9036 un poco in avanti a partire da $8EF0. Questo si ottiene facilmente; entriamo di nuovo in monitor con CALL-151 e scriviamo:
8EF0<8F3B.9036M (Return)
Torniamo in Applesoft con 3D0G (Return) e salviamo tutto il programma con il nome REN-.CODE:
BSAVEREN.CODE,A$8800,L$800 (Return)
Ora carichiamo il programma del ROMWRITER, selezioniamo l'indirizzo di partenza come $C800 e quello finale come $CFFF. Quando l'Apple ci chiede se vogliamo caricare un programma, rispondiamo con «Y» e scriviamo REN.CODE. Poi bisogna dare l'O.K. alla programmazione rispondendo di nuovo «Y» alla domanda «ARE THE ADDRESSES CORRECT Y/N?»
Dopo la programmazione bisogna rimettere l'HIMEM a $9600 scrivendo:
HIMEM:38400 (Return)
Inoltre bisogna porre il deviatore CFFF on/off in posizione ON, cioè in giù. Conviene anche mettere il deviatore di Write Protect in su per evitare cancellazioni accidentali.
A questo punto possiamo, finalmente, cominciare ad usare il Renumber. Premiamo PR#N dove N, come al solito, è lo slot. Con un programma in memoria basta premere & per renumerare il programma a passi di dieci.
Inoltre vengono eseguiti tutti i comandi elencati nelle istruzioni del Renumber che permette di rinumerare a passi diversi da dieci, rinumerare solo una parte del programma, cucire insieme due programmi ecc.

Conclusione

Il ROMWRITER è facilissimo da usare e, per chi traffica con hardware e programmi in linguaggio macchina, è uno strumento praticamente indispensabile. La costruzione della scheda è ottima: vetronite a doppia faccia con fori metallizzati, connettore dorato e zoccoletto ZIF di ottima qualità, come già descritto. Il costo in Italia di questa scheda è forse, come per la maggioranza delle periferiche, un po' alto: negli stati Uniti il ROMWRITER costa 159 dollari...

Bo Arnklit

@ prove #

CBM COMMODORE: L'INACCESSIBILE COMPUTER.

Accessibile nel prezzo, accessibile nel linguaggio.

Per professionisti, aziende, tecnici, appassionati oggi c'è CBM Commodore, il microcomputer accessibile a tutti. Accessibile nel linguaggio perchè si programma in Basic, il più semplice, e bastano poche ore per imparare ad usarlo. Accessibile nell'utilizzazione, perchè può essere applicato a risolvere un'infinità di problemi e installato ovunque. Accessibile nel prezzo: molto contenuto, rispetto alle prestazioni offerte.

Fornito, programmato, assistito.

Da ABA Elettronica, un'azienda specializzata e rappresentante ufficiale. I suoi tecnici sono a disposizione del cliente per studiarne tutti i problemi, per realizzare e implementare i programmi necessari, per offrire un'assistenza completa hardware e software.
Insomma, con CBM Commodore il microcomputer è davvero diventato accessibile. A tutti.

Un microsistema completo e affidabile.

Con 8 - 16 - 24 - 32 K RAM, video display, floppy disk a doppio drive, una vasta gamma di stampanti, plotters, interfacce e tutte le apparecchiature IEEE 488 compatibili (tutta la strumentazione Hewlett Packard, ad esempio), CBM Commodore è un sistema veramente completo. E può risolvere problemi di ogni tipo: gestionale, tecnico, scientifico, statistico, finanziario, garantendo sempre la massima affidabilità, sperimentata in migliaia di applicazioni.

Distribuzione, Programmazione e Assistenza: ABA ELETTRONICA - 10141 Torino via Fossati 5/c - Tel. (011) 332065 - 389328
Importatrice per l'Italia: HARDEN S.p.A. - Sospiro (CR)

LA CRITTOGRAFIA COMPUTERIZZATA

di Filippo Merelli

Nelle applicazioni che richiedono la disponibilità di archivi o messaggi riservati (in altre parole segreti), l'uso di un calcolatore può dare ampie garanzie sia di praticità di codifica e decodifica, sia di sicurezza, perché é possibile utilizzare algoritmi anche molto complessi. I requisiti ai quali l'algoritmo deve rispondere sono più numerosi di quanto si possa credere: dall'identificazione dell'operatore all'autodifesa del sistema contro tentativi di forzatura, al riconoscimento di testi falsi o alterati, alla necessità che l'algoritmo stesso vari nel corso della codifica. L'articolo analizza il problema nel suo insieme; nel prossimo numero verrà presentato un esempio di programma di codifica e decodifica implementato su HP 41 C.

L'esigenza di poter disporre di sistemi di comunicazione non inquinabili, protetti da eventuali intrusioni e soprattutto sicuri è sentita in diversi campi d'applicazione: tra questi citiamo il trasferimento di fondi e la gestione a calcolatore dei depositi bancari, le comunicazioni via satellite, la gestione di archivi riservati e tutti gli altri casi in cui una certa risorsa viene condivisa tra più utilizzatori.
Si consideri ad esempio la gestione di uno studio di consulenza fiscale ed i problemi che sono connessi con l'utilizzazione dei dati disponibili in questa attività.
A ciò si è cercato di rispondere con l'utilizzazione di cifrari che codifichino il messaggio originario, o sorgente (MS) in un messaggio codificato (MC) per quanto possibile non alterabile da parte di persone non autorizzate e che contemporaneamente fornisca buone garanzie di non rivelare, oltre al messaggio stesso, anche il cifrario usato, ossia la «chiave» per interpretare altri messaggi (o dati) provenienti dalla stessa fonte.
Vogliamo comunque sottolineare come l'esigenza della protezione crittografica non derivi esclusivamente dal voler mantenere riservato il contenuto di determinati testi, ma sia dovuta a necessità di sicurezza intrinsecamente collegate con l'espansione delle attività gestite tramite calcolatore.
Una semplice translitterazione (ossia il sostituire una lettera del MS con un'altra o con una opportuna cifra in modo da formare una corrispondenza biunivoca MS ⟷ MC) non risolve il problema in maniera adeguata.
Infatti, dall'osservazione della percentuale d'incidenza della varie componenti del MC (percentuale tipica della lingua usata nella stesura del messaggio) si può risalire anche se con qualche difficoltà nel caso di testi molto brevi, al MS.
Tralasciamo qui da analizzare i metodi legati a linguaggi figurati o metaforici e quelli cosiddetti «a crivello» (in cui si forma una «maschera» contenente il MS e si riempiono gli spazi vuoti con parole per quanto possibile attinenti logicamente alle altre) in quanto essenzialmente empirici e quindi con affidabilità non quantificabile.
Tali metodi, tra l'altro, non si prestano alla trattazione di dati essenzialmente numerici e non sono praticamente adoperabili fuori del

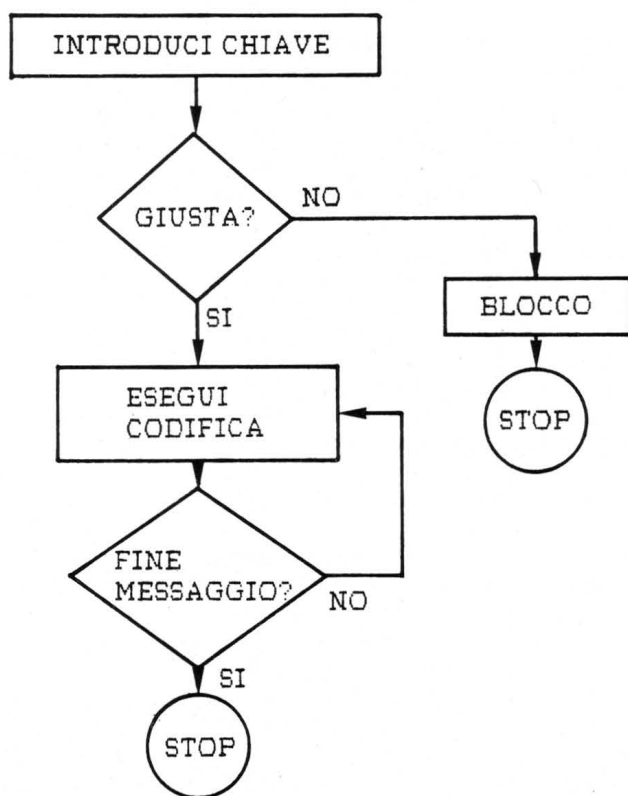
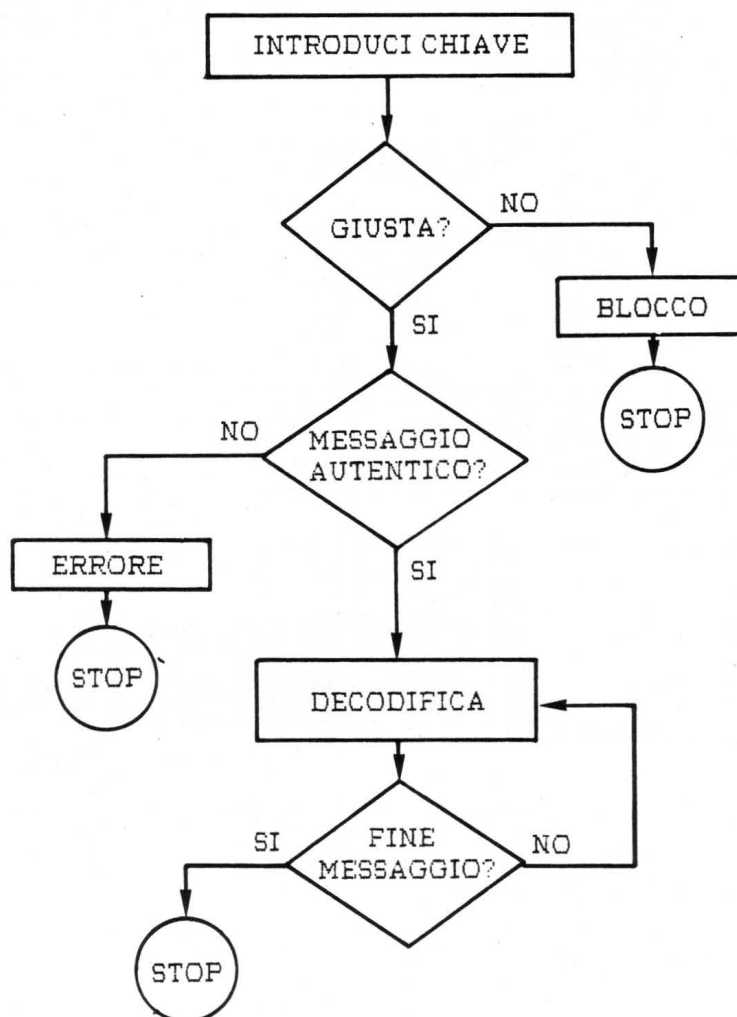



Fig. 1: «Secret» schemi di principio

settore «classico» dello spionaggio.
Non bisogna inoltre dimenticare che l'analisi di un MC può oggi essere condotta con l'ausilio del calcolatore, cosa che riduce sensibilmente i margini di sicurezza offerti dai vari sistemi di crittografia finora in uso.

Requisiti

Diverse sono le specifiche cui deve soddisfare un sistema «di sicurezza» a seconda che sia destinato prevalentemente a comunicazioni tra utenti di una stessa rete o venga utilizzato in applicazioni «statiche» quali ad esempio la creazione di archivi riservati o banche dati.
Nel primo caso bisognerà prevedere il verificarsi di possibili alterazioni del messaggio dovute a fattori esterni (intenzionali o non): il codice usato dovrà quindi presentare un certo grado di ridondanza che permetta (in sede di ricezione) di autocorreggere (entro certi limiti) o quantomeno di segnalare eventuali errori.
Nel secondo caso ci si può limitare alla codifica, dovendocisi assicurare solo contro eventuali «letture» anomale del messaggio; parallelamente però cresce l'importanza di poter disporre di una buona codifica, essendo i testi protetti invariabili nel tempo.
Bisogna poi che il sistema usato sia in grado di riconoscere l'operatore, eventualmente con una certa scala di priorità che consenta l'esecuzione solo di determinate operazioni (codici a protezione multipla).
Doverosa segnalazione merita la possibilità di poter riconoscere l'autenticità di un messaggio codificato mediante un codice di autenticazione (MAC) contenuto nel messaggio stesso, al fine di garantirsi contro eventuali sostituzioni di testi.
Ultima (non certamente per quanto riguarda l'importanza che riveste) è la possibilità del sistema di resistere ad eventuali tentativi di forzatura da parte di persone non autorizzate, di cui i più pericolosi sono quelli di tipo esaustivo (midnight attack).

Codificatori elettronici

L'avvento dell'elettronica programmabile sul larga scala ha offerto un modo nuovo e all'altezza dei tempi per risolvere in maniera definitiva il problema delle comunicazioni «sicure».
Disponendo infatti della potenza di elaborazione offerta dai moderni calcolatori si sono potuti sviluppare algoritmi di codifica che offrono una protezione assoluta contro tutti i possibili attacchi esterni.
È quanto vogliamo dimostrare con il tipo di codificatore, realizzato a calcolatore, presentato in queste pagine.
Oltre ad offrire una inattaccabilità assoluta ed una affidabilità estremamente elevata (pari a quella del calcolatore usato), il sistema crittografico «SECRET» presenta le seguenti caratteristiche peculiari:
— funzionamento del codificatore e (soprattutto) del decodificatore abilitato soltanto da parte di persone autorizzate, mediante un'apposita combinazione utilizzata come «chiave elettronica»; tale combinazione può essere personalizzata per ogni singolo utente che ha accesso alla codifica.
— I simboli costituenti il MC, pur essendo in

corrispondenza quantitativa 1:1 con i simboli del MS, sono costituiti da numeri aventi distribuzione pseudo-casuale, rendendo quindi impossibile l'eventuale analisi logica del testo codificato al fine di ricavare il codice utilizzato.

— Il codice stesso è inoltre variabile, e da messaggio a messaggio e nel corso del messaggio stesso, rendendo quindi vano il lavoro di descrittazione svolto su messaggi pervenuti a persone non autorizzate.

Ricostruire il MS partendo dal MC è comunque IMPOSSIBILE, qualsiasi sia la lunghezza del testo.

— Il codice utilizzato presenta in uscita i dati costituiti da blocchi di lunghezza fissa, contribuendo ulteriormente alla sicurezza del sistema.

Il sistema crittografico «SECRET» è per sua natura del tipo «custom» e può quindi essere usato in più applicazioni senza reciproche interferenze conservando tutti i requisiti di sicurezza con minimi cambiamenti software.

Pur essendo stato sviluppato essenzialmente per la protezione delle informazioni di archivi riservati (off line storage), il sistema «SECRET» si presta ad essere usato anche in applicazioni di messaggistica, essendo dotato del controllo di autenticità già richiamato (MAC). Inoltre eventuali errori sul singolo carattere (disturbi ecc.) non provocano alterazioni nei caratteri adiacenti.

Fra le caratteristiche operative c'è da segnalare la «protezione» dei programmi di codifica e di decodifica, che non possono quindi essere letti od analizzati.

Inoltre l'introduzione nella macchina di una «chiave» errata provoca il blocco dei programmi stessi e la loro eventuale cancellazione, in modo da impedire che eventuali attacchi di tipo esaustivo possano in qualche modo portare ad un «forzamento» delle misure di sicurezza all'ingresso del sistema (autodifesa).

Flow chart

Vediamo ora come sono organizzati i flow chart delle parti codifica e decodifica relativamente al programma principale (fig. 1).

La parte centrale di tutto il programma di crittografia è quindi rappresentata dal rettangolo (task) contrassegnato «Esegui codifca», che esamineremo ora in dettaglio.

Esso è basato sulla corrispondenza biunivoca tra una lettera dell'alfabeto (es. B) e il suo numero progressivo (es. 2), modificata però dall'aggiunta di un numero generato da un generatore casuale, di modo che la stessa lettera (o cifra o carattere speciale) dia origine a codifiche sempre differenti.

In sede di decodifica poi vi è un identico generatore casuale ed il programma provvede al ripristino della lettera originaria.

Per mantenere il sincronismo tra i due generatori viene usata una inizializzazione diversa per ogni messaggio (seme) anch'essa adeguatamente protetta.

L'identificazione dell'operatore è assicurata da una o più chiavi elettroniche che possono essere di due tipi:

1) personalizzate ed assegnate arbitrariamente, in modo che ogni utente abbia accesso ad una sola macchina;

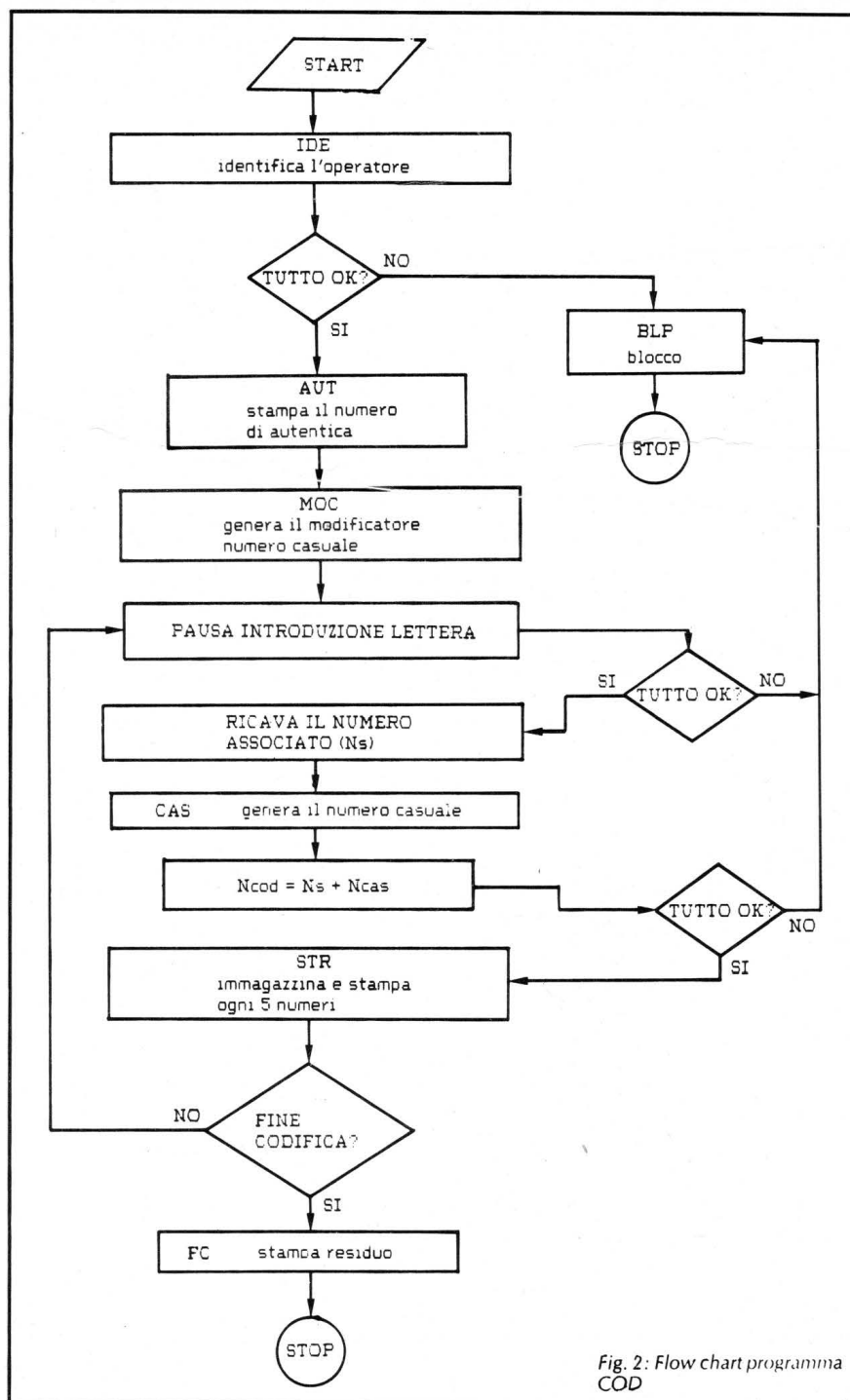



Fig. 2: Flow chart programma COD

2) personalizzate per ogni utente ma formate da 2 parti collegate da opportuna funzione segreta, per consentire a più utenti di utilizzare la stessa macchina pur mantenendo la diversità delle chiavi.

In fig. 2 è riportato il flow chart della parte codifica (COD) con l'indicazione delle funzioni logiche eseguite.

In fig. 3 viene riportato il flow chart del programma di decodifica (DEC).

Descrizione delle funzioni

L'intero sistema di crittografia «SECRET» è stato strutturato in maniera modulare, in cui ogni signola funzione logica del programma viene identificata ed eseguita per mezzo di sobroutine del programma principale, che costituisce così lo «scheletro» dell'elaborazione.

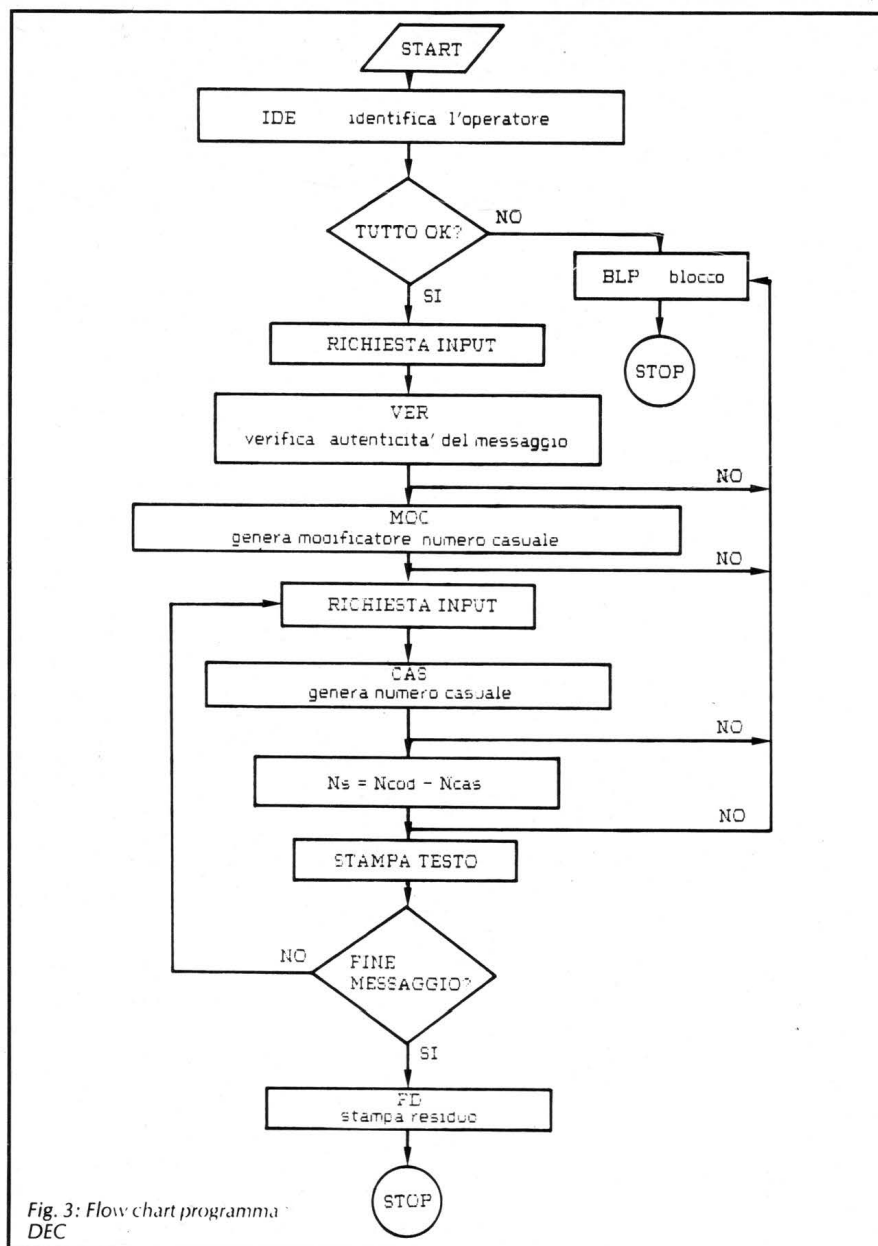



Fig. 3: Flow chart programma DEC

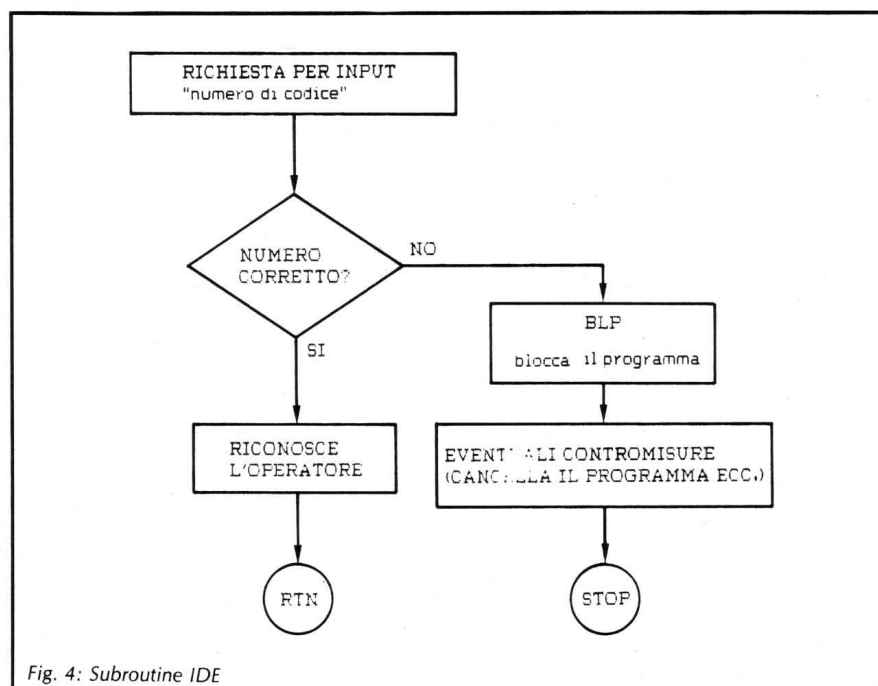



Fig. 4: Subroutine IDE

Questo approccio al problema permette, da un lato, un editing più veloce e sicuro, dovendosi provare dei segmenti di programma di limitata complessità e parimenti facilita l'implementazione del sistema su macchine aventi architetture differenti.
Tutto ciò porta, l'altra parte, ad una esecuzione leggermente più lenta del programma principale, dovendo ogni volta chiamare le subroutine utilizzate, e ad una utilizzazione non ottimizzata delle risorse della macchina impiegata.
Personalmente ritengo che la tecnologia attuale consente di adoperare senza conseguenze negative apprezzabili (salvo casi eccezionali) questa impostazione più lineare e più pratica dei problemi di programmazione, soprattutto quando si prevedono possibili modifiche concettuali ai programmi o ai blocchi sviluppati.
Ciò premesso vediamo ora le subroutine usate nel programma di codifica (COD).
1) IDE = identifica l'operatore.
Costituisce la «chiave di accesso» al programma di codifica (e di decodifica) e come tale può essere specializzata nel compiere operazioni differenti a seconda della macchina usata.
Inutile sottolineare come una debolezza strutturale in questo punto di un sistema crittografico possa portare a risultati fatali.
Lo schema logico prescelto per questo compito è riportato in fig. 4.
Con la dizione «numero di codice» la macchina può richiedere l'input di dati personali, combinazioni alfanumeriche ecc. (eventualmente possono essere usate più chiavi).
A questo proposito bisogna tenere presente che complicare eccessivamente la procedura di ingresso porta ad una diminuzione del grado di sicurezza offerto dal sistema, stante l'impossibilità di ricordare mnemonicamente un numero eccessivo di combinazioni.
D'altro canto esigenze di «protezione» del programma consigliano di utilizzare chiavi che rappresentino punti singolari di funzioni matematiche prefissate.
Il segmento BLP provvede a bloccare il programma ed è stato etichettato per poter essere utilizzato anche in altri punti del programma, ove sono previsti dei test del tipo «tutto OK?» per impedire un accesso non sequenziale alle parti interne del codificatore.
Le contromisure indicate nel flow chart permettono di realizzare, anche attraverso eventuali modificazioni del programma memorizzato in macchina, l'autodifesa del sistema cui si è già accennato.
2) AUT = stampa il numero di autentica del messaggio.
Questa subroutine è costituita da tre blocchi funzionali:
— il primo comprende un generatore indipendente di numeri casuali, utilizzato come riferimento «interno» della macchina;
— il secondo genera il codice di autentica vero e proprio (MAC) partendo dal numero casuale di cui al punto precedente tramite una funzione a più valori, in modo da non formare una corrispondenza biunivoca numero⟷risultato, che potrebbe portare alla identificazione della funzione stessa.

Può essere adoperata, ad esempio, una funzione del tipo:

$$Y = \frac{\text{Sen } X + \text{Cos } X}{|^{\lg(\text{sen } x/2)}}$$

in cui X è il numero casuale.
Tale operazione viene eseguita da un'apposita subroutine denominata FUN.
Opportuno è anche procedere ad un troncamento delle cifre più significative della funzione stessa:
— il terzo blocco manipola in modo opportuno la quantità numero + risultato, in modo da proteggere ulteriormente la autenticazione del messaggio. Ad esempio può essere usata una scalatura a chiave del tipo:

AA,BBB → CC,DDD → DDAACCBBDB

num. casuale (x) risultato (y) num. di autentica N_A

Con queste supposizioni ecco quindi il folw chart della subroutine AUT (fig. 5).
Il numero di autentica (nella forma codificata N_A) rappresenta il primo numero di 10 cifre costituente il messaggio codificato; il suo «aspetto esteriore» è del tutto simile a quello dei blocchi di 10 cifre successivi.
L'analisi della quantità N_A (formata dalle parti indipendenti AA,BBB e CC,DDD) permette, in sede di decodifica, di stabilire se un determinato messaggio è autentico o meno.
Per quanto riguarda il generatore di numeri casuali può essere utilizzata con successo una formula interativa del tipo (ad esempio):

$$N_i = \text{FRAC}\left[(N_{i-1} + k)^{3.41}\right]$$

in cui la quantità cercata viene ricavata dalla parte frazionaria dell'espressione tra parentesi quadre, dove compare la quantità del passo precedente.
La costante k è un numero arbitrario di grandezza comparabile con N_i.
A seconda del campo di variaziione ammesso per la variabile casuale si procede poi ad uno shift verso destra dei posti necessari per l'applicazione interessata.
In questo caso la quantità AA,BBB viene ricavata mediante shift di 5 posizioni e successiva divisione per 1000 della quantità N_i.
Un generatore casuale basato sullo stesso principio viene utilizzato anche nella subroutine CAS: per questa applicazione l'uscita deve essere limitata ad un numero di 2 cifre, per cui vanno opportunamente adeguate le istruzioni di shift.
Questi generatori casuali superano il test spettrale di correlazione e risultato quindi adatti allo scopo; nulla vieta tuttavia di utilizzarne altri basati su principi diversi.
La sequenza di numeri casuali generata è condizionata dalla inizializzazione del generatore, effettuabile tramite la grandezza N_{i-1}, a meno che si tratti come in questo caso di generatori del tipo free running, il cui funzionamento è svincolato da esigenze di sincronizzazione.
3) MOC = genera il modificatore casuale.
Scopo di questa subroutine è la generazione di un «seme» per l'inizializzazione del generatore di numeri casuali utilizzato nella codifica (subroutine CAS). Al fine di ottenere codifiche sempre differenti viene utilizzato ogni volta un seme diverso, ricavato con una opportuna manipolazione del numero di autentica in uscita dalla subroutine AUT.

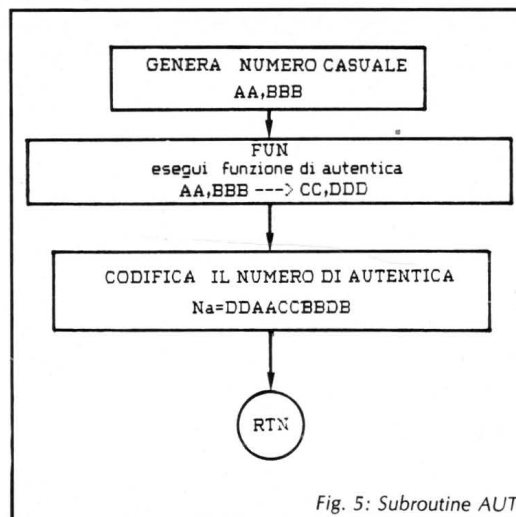
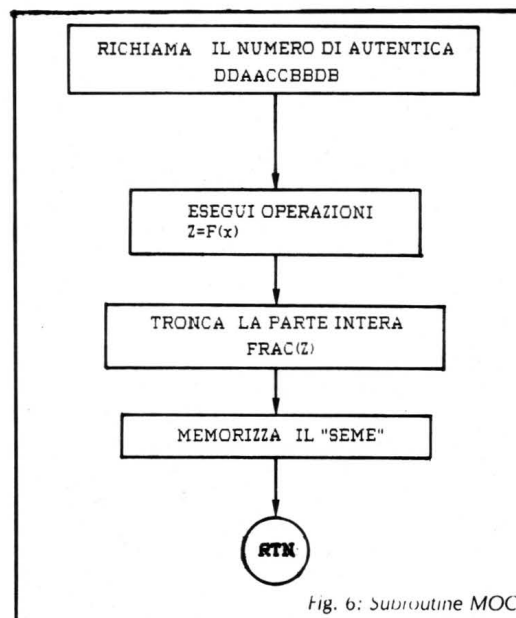
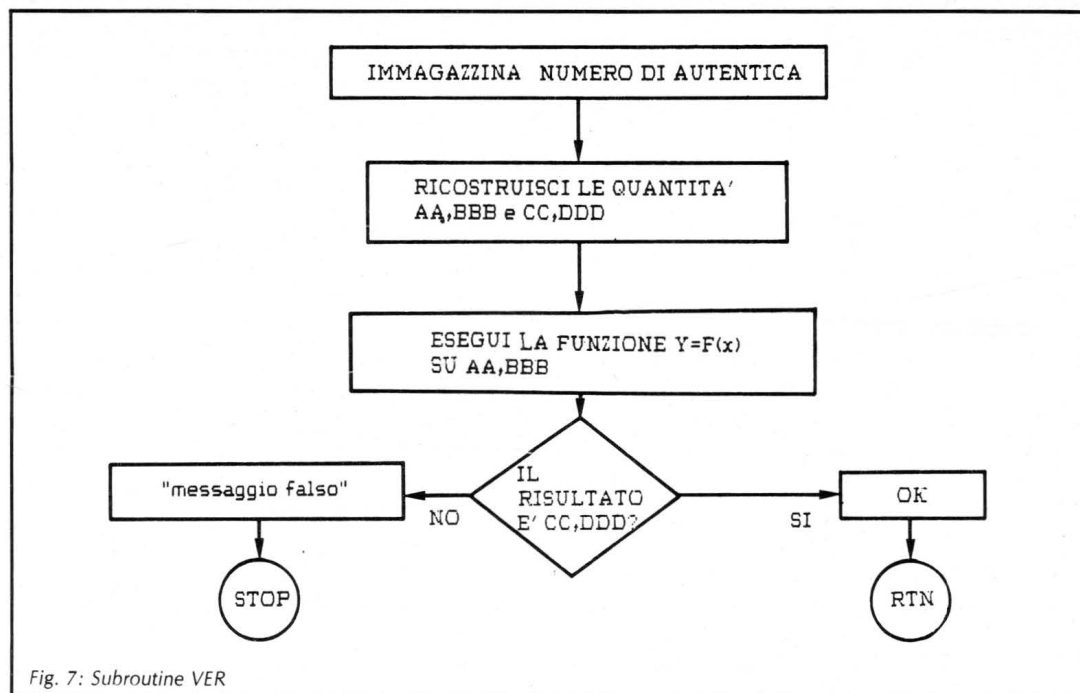



Fig. 5: Subroutine AUT

Fig. 6: Subroutine MOC

Per la funzione z=F(x) valgono le considerazioni già fatte a proposito della subroutine AUT sulle funzioni a più valori.
4) CAS = genera il numero casuale.
Viene utilizzata per generare il numero pseudo-casuale di inserire nella codifica in modo da aversi $N_{COD} = N_S + N_{CAS}$ in cui :
N_{COD} è il numero corrispondente alla lettera (o carattere sorgente) codificata.
N_S è il numero associato (arbitrariamente) al singolo carattere dell'alfabeto sorgente (si tratta quindi di una translitterazione).
N_{CAS} è il numero generato dalla subroutine CAS.
In proposito valgono le seguenti considerazioni:
— data la dimensione D_S dell'alfabeto sorgente che si vuole codificare (ad esempio D_S = 26 lettere + 10 cifre + n caratteri speciali) si può scegliere il numero di cifre da utilizzare per la codifica del singolo carattere (C_{COD}).
— La dimensione del generatore casuale D_{CAS} deve essere comparabile con la grandezza D_S, al fine di ottenere in uscita un numero ancora pseudo-casuale.

Fig. 7: Subroutine VER

Ad esempio avendo un alfabeto da codificare di 36 caratteri (D_S = 36) il numero di cifre minimo per ogni singolo carattere è C_{COD} = 2. Con tale scelta il generatore di numeri casuali deve essere limitato per fornire cifre ≤ 63, in modo che la quantità N_{COD} sia ancora rappresentabile con un numero di 2 cifre.
Tale implementazione, pur nella sua essenzialità, dà risultati senz'altro soddisfacenti.
Codificando invece con 3 cifre per carattere (C_{COD} = 3) si hanno a disposizione posizioni non utilizzate che possono essere sfruttate per dotare il sistema di un certo grado di ridondanza in modo da renderlo adatto ad applicazioni di messaggistica, in cui il testo ricevuto può risultare alterato da cause accidentali.
Del pari può essere aumentato il limite superiore D_{CAS} del generatore casuale. 5) STR = immagazzina e stampa ogni 5 numeri.
Scopo di questa sobroutine è di formattare il messaggio codificato in modo da presentarlo in blocchi di lunghezza fissa ed agevolmente reimpostabili in sede di decodifica. Il compromesso più opportuno si è trovato essere costituito da blocchi di 10 cifre, il che, nel caso della codifica minima vista precedentemente, consente di raggruppare 5 caratteri alla volta.
Nel caso che l'output del testo codificato non sia su stampante (ma ad esempio venga archiviato su cassetta o diskette) questa sobroutine può essere sostituita con un'altra di tipo più appropriato fermi restanto i vantaggi (teorici e pratici) offerti dalla segmentazione in blocchi di lunghezza fissa. 6) FC = fine codifica.
In questa sezione viene stampato il residuo del registro di uscita con i numeri eventualmente contenuti; viene inoltre aggiunto il controllo di «fine messaggio».
Vediamo ora di esaminare le caratteristiche del programma di decodifica (DEC).
Essendo parte dello stesso sistema di crittografia, il programma DEC utilizza alcune funzioni comuni al programma di codifica (subroutine IDE, MOC, CAS) che vengono semplicemente richiamate nel corso dell'esecuzione; le funzioni da costruire «ex novo» sono riportate di seguito.
7) VER = verifica autenticità del messaggio.
Scopo di questa sobroutine è di verificare che l'input della macchina sia effettivamente costituito da un messaggio autentico (fig. 7).
Dopo aver ricostruito le quantità AA,BBB e CC,DDD partendo dal numero di autentica (nella forma DDAACCBBDB) viene eseguita la funzione segreta y = F(x) (per mezzo della subroutine FUN) sulla quantità AA,BBB. Se il risultato di questa operazione è effettivamente CC,DDD viene abilitato il seguito del programma, avvisando contemporaneamente l'operatore. 8) FD = fine decodifica.
Viene stampato il residuo (frazione di parola) presente nei registri di uscita e si controlla la presenza del carattere di «fine messaggio» (EOF).
In sede di decodifica vi è anche da tenere conto della formattazione usata, per poter mantenere il passo tra parola codificata e parola decodificata (dimensionamento appropriato del loop di parola).
Aggiungiamo delle osservazioni importanti di carattere generale:
1) uno stesso messaggio, codificato dalla stessa persona in tempi diversi, dà origine a codifiche differenti (sicurezza assoluta).
2) Non si può utilizzare il numero di autentica del messaggio (o il MAC) separatamente dal testo seguente, cioè il messaggio può essere riconosciuto autentico dal sistema solo in presenza del suo MAC (garanzia contro le falsificazioni).
3) Le chiavi di accesso al sistema crittografico sono indipendenti dal testo codificato, ovvero è possibile cambiare la procedura di riconoscimento e di abilitazione dell'operatore senza conseguenze sul testo codificato (e sugli altri testi già archiviati). A questo punto, il lettore dispone di tutti gli elementi per implementare il programma SECRET sul proprio calcolatore.
A titolo di esempio, presenteremo, nel numero prossimo, una versione per HP 41 C. ■

S10 e S30

stampanti Honeywell per mini-micro e professional personal computer

S10: 80 colonne S30: 132 colonne

• Tecnologia matrice a impatto • 80 CPS con stampa bidirezionale • Interfaccia seriale fino a 9600 bps e parallela • Alta qualità di stampa • Self test • Set di 64/96 caratteri

ANCHE NELLE STAMPANTI È MEGLIO HONEYWELL

prodotti O.E.M.

Honeywell

Honeywell Information Systems Italia

Filiale O.E.M. - Via Tazzoli 6. 20154 Milano. Tel. (02) 6570312 - 69771 - Telex 311308 HISI

S.O.A. note

di Pierluigi Panunzi

Per quanto riguarda le etichette, oltre alle 72 previste, possono essere usate anche le seguenti, il cui codice è indicato tra parentesi: Pgm Ind (62), Exc Ind (63), Prd Ind (64), STO Ind (72), RCL Ind (73), SUM Ind (74), HIR (82), GTO Ind (83), Op Ind (84), INV SBR (92).

L'istruzione HIR (codice a due byte 82 XY) consente la gestione dei registri usati dal Sistema Operativo Algebrico (gli ultimi 4 dei quali usati anche per la stampa), così come se fossero dei registri «normali». In pratica le cifre del secondo byte indicano qual'è l'operazione da compiere e su quale registro (Y=1,2,...,8). Ad ogni valore della cifra X corrisponde un'operazione secondo la tabella:

valore di X	corrisponde a
0	STO
1	RCL
2	Nop
3	SUM
4	Prd
5	INV SUM
6,7,8,9	INV Prd

Bisogna fare attenzione quando si effettuano operazioni (con il valore della cifra X maggiore di 2) con numeri in valore assoluto minore di 1 per i quali è necessario usare la notazione esponenziale (*EE* o *Eng*), altrimenti il numero viene alterato.

Torniamo sull'argomento della compatibilità delle TI-59 nella lettura di schede magnetiche: ebbene nei manuali «vecchi» non era segnalato alcunché in merito, lasciando presupporre per il meglio. Solo nei manuali «nuovi» è riportata (... «a grandi lettere»... come dice il lettore Fazzi nella Posta del n° 6) un'aggiunta in cui è detto che non si garantisce la compatibilità. In effetti parecchi lettori hanno inviato insieme al programma le relative schede magnetiche e (fortunatamente per noi!) nella maggioranza dei casi risultavano perfettamente leggibili.

Supponiamo di avere una certa sequenza di istruzioni alla fine della quale c'è il test stu di un *flag*. A seconda del suo stato (on/off) il programma procederà da una parte (on) o da un'altra (off). Il nostro problema è di controllare questa diramazione premendo rispettivamente *A* oppure *B*, facendo eseguire in entrambi i casi la sequenza: ciò è facilmente realizzabile con le istruzioni *Lbl A INV Lbl B St flg 0... (sequenza)...(test sul flag 0)...*

Premendo *B* il flag 0 viene settato (on) mentre premendo *A* il flag viene resettato (off): può sembrare strano l'uso dell'*INV* seguito da *Lbl* quasi come se volessimo l'«inverso di un'etichetta»; in realtà l'*INV* non risente della presenza della *Lbl B* e tutto va come se ci fosse stato *INV St flg 0* (ciò è infatti confermato usando la stampante in «*trace*»).

Questa possibilità di separare l'*INV* dalla funzione considerata tramite un'etichetta è valida in generale per tutte le funzioni di cui è lecita l'*inversa* (vedi pag. V-4 del manuale, ad eccezione di *SBR* e di *Fix*). Non sfugge da questa regola nemmeno la HIR però solo nel caso in cui il secondo byte è del tipo 3Y o 4Y (rispettivamente equivalenti a SUM E prd); ad esempio la sequenza *INV HIR 35* equivale a sottrarre il numero dato al contenuto del registro interno n° 5 (come se avessimo fatto *HIR 55*). Tra l'altro, con la stampante in «*trace*», in questo caso si ha «*IHIR*».

Abbiamo riscontrato, in media, che i modelli più recenti delle tre calcolatrici, diciamo costruite nel 1979-80, sono molto più veloci delle loro sorelle «più anziane», intendendo con questo che uno stesso programma viene eseguito in tempi nettamente inferiori). Un caso anomalo è stato riscontrato in una «vecchia» TI-58 che all'improvviso è diventata più veloce!

Altro spunto riguarda il Pgm 07 che permette la valutazione di polinomi per valori qualsiasi della variabile *x*. Se i coefficienti del polinomio sono calcolati dal nostro programma (vedi ad esempio «Interpolazione di Stirling» del n° 4) allora basta porli in successione nei registri dallo 05 in poi. Dopo aver memorizzato il grado del polinomio in STO 04 (cioè *senza* effettuare la sequenza *Pgm 07 A*, con i vantaggi già visti) e richiamato il valore della variabile x, basta semplicemente inserire la sequenza *Pgm 07 C*, la quale utilizza anche i registri 01, 02, 03.

Esiste un'ulteriore possibilità di accedere ad una certa parte di un programma di libreria, *non tramite etichette*, ma bensì con indirizzamento assoluto. Ciò può al limite essere utile in particolarissimi casi, in cui una certa sequenza è «identica» alla parte finale di un programma di libreria (che termina cioé con *R/S* o *RTN*). Ad esempio supponiamo che nel nostro programma dobbiamo controllare se la memoria 05 contiene il valore *100*: in caso negativo dobbiamo fermare l'elaborazione, segnalando errore (con 9.9999999 99 lampeggiante) altrimenti proseguiamo. Il frammento di programma potrebbe essere questo: *...100 x ⇄ t RCL 05 x=t RCL 0 1/x R/S Lbl RCL...*

Ora però possiamo sfruttare i passi 114, 115, 116 e la label *x* del Pgm 17 ed allora la sequenza sarebbe

...100 x ⇄ t RCL 05 Pgm 17 SBR 114...

Evidentemente non sono interessanti i due passi in meno (anzi è abbastanza scomodo trovare sequenze da poter usare), ma è importante conoscere l'esistenza di quest'altro tipo di accesso ai programmi di libreria: ovviamente se l'indirizzo assoluto usato supera l'effettiva lunghezza del Pgm richiamato, la calcolatrice segnala errore.

Veniamo a spunti, dettati dall'esperienza nell'uso dei programmi del modulo Master, che consentono di accorciare di un certo numero di istruzioni i nostri programmi. Supponiamo ad esempio che si debba generare una sequenza di numeri casuali interi compresi fra 0 e 10, usando come numero di «innesco» 0.5. Seguendo il manuale d'uso del modulo, dovremmo, nel programma, porre la sequenza *Pgm 15 E* per introdurre il numero d'innesco e per il resto fare:

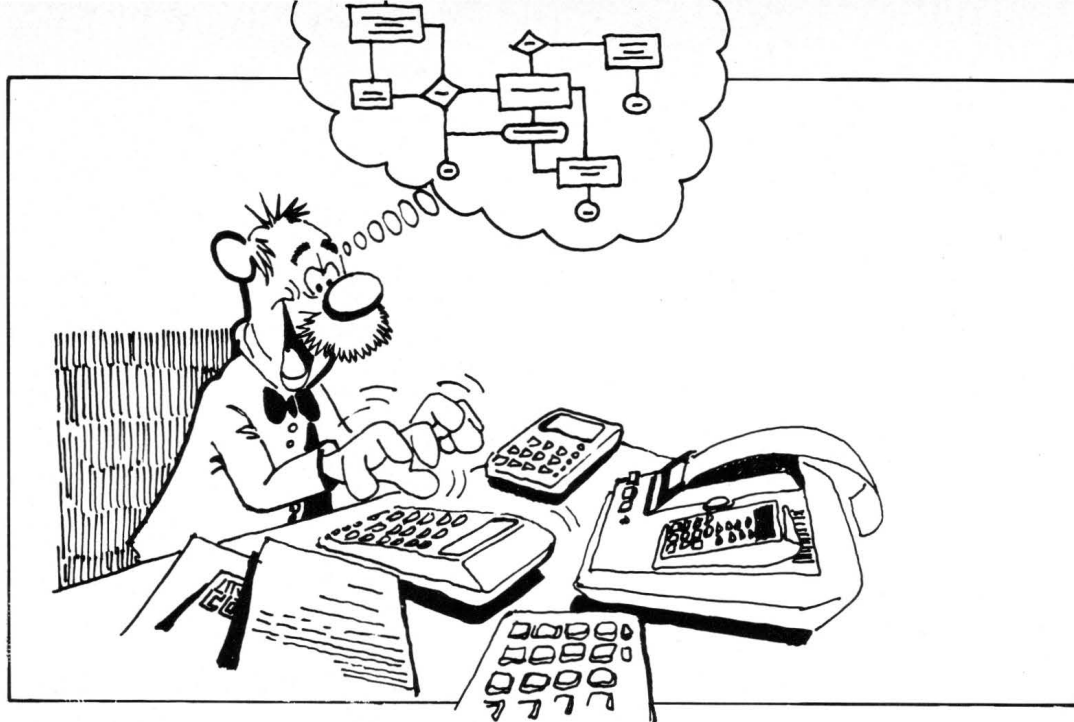
0 Pgm 15 A 1 0 Pgm 15 B... Pgm 15 C Int

dove le ultime quattro istruzioni saranno in un ciclo. In tutto usiamo 3+9+4=16 byte con il grave handicap che il Pgm 15 usa (come riporta il manuale) i registri 01, 02,..., 07, 09, con la parte etichettata *C* che da sola, oltre a generare il numero con i requisiti richiesti, calcola la media e la varianza dei numeri usciti volta per volta, usando i registri «statistici» 01, 02,..., 06. Ora tutto ciò può essere accorciato andando a vedere cosa fanno in realtà le parti etichettate con *E,A,B*: queste consistono soltanto nella memorizzazione del numero presente all'atto della chiamata nei registri 09, 10, 11 rispettivamente. La sequenza può diventare: per la prima parte *STO 09* (già un byte risparmiato ed evitata una chiamata a subroutine con conseguente risparmio di tempo) e per la seconda parte *0 STO 10 1 0 STO 11 ... Pgm 15 C Int...* per un totale di 2+7+4=13 byte, oppure ancora meglio *Pgm 15 SBR D.MSx10=Int* per un totale di soli 2+9=11 byte.

L'istruzione *Dsz*, che gestisce i *loop* controllando il numero di iterazioni in base al contenuto di un registro, può essere estesa ad un qualsiasi registro. Infatti, contrariamente a quanto cita il manuale a pag. V-63, nel secondo byte dell'istruzione (il numero del registro) si può porre un numero di due cifre relativo ad una qualsiasi cella di memoria effettivamente presente per la ripartizione adottata e che fungerà da contatore nel loop. È da notare che il numero di due cifre (NN) non si può inserire direttamente da tastiera, ma con un artificio, ad esempio introducendoli con RCL NN ed eliminando la RCL, sfruttando opportunamente le istruzioni *SST, BST, Ins, Del*.

Unico registro che non può essere usato in questa estensione è il n° 40 in quanto la sequenza 97 40 ecc. è considerata come *Dsz Ind ecc.* cioè si riferisce al controllo di cicli in cui il contatore è gestito indirettamente.

Software S.O.A.

a cura di Pierluigi Panunzi

Inviate a m&p COMPUTER i vostri migliori programmi in S.O.A. (Sistema Operativo Algebrico per calcolatrici Texas Instruments). Saranno esaminati dalla Redazione; i più interessanti verranno pubblicati e gli autori ricompensati con un modulo Solid State Software a loro scelta, fra quelli disponibili nel catalogo Texas Instruments: Statistica applicata - Aviazione - Navigazione marina - Decisioni in affari - Immobili/investimenti - Analisi dei titoli finanziari - Matematica/uso della stampante - Simulatore RPN - Agraria - Ingegneria elettronica - Ingegneria civile - Topografia - Analisi per il trattamento delle acque. Si prega di inviare il materiale nella forma più ordinata possibile, scrivendo a macchina o comunque in maniera ben leggibile, e documentando i programmi con spiegazioni, commenti ed esempi.

Prima di analizzare il programma premiato questa volta, facciamo il punto della situazione: dalle lettere pervenuteci (e in ogni caso archiviate, mai cestinate!) oltre a selezionare gli elaborati da premiare abbiamo raccolto anche parecchi «trucchi programmativi», estensioni varie e curiosità, riguardanti le calcolatrici TI - 58, 58C, 59. Pensiamo perciò di far cosa gradita ai lettori dedicando ampio spazio a queste notizie, alcune delle quali già segnalate in numeri precedenti, insieme ad altre nuove di zecca.
Vediamo però prima il programma premiato: riguarda la soluzione di un sistema di n equazioni lineari in n incognite. C'è da dire che nel modulo Master già esiste un programma simile e perciò approfittiamo di questo nuovo programma per presentare un altro metodo risolutivo, quello dell'eliminazione diagonale o metodo di Gauss-Seidel, insieme ad alcune considerazioni sulla gestione delle matrici. Lasciamo dunque la parola al vincitore: si tratta di Enrico Andrianopulos di Casalecchio di Reno.

Risoluzione del sistema di equazioni lineari con il metodo di eliminazione diagonale

Consideriamo il sistema di n equazioni lineari nelle n incognite X_i (i=1,2, , ,n)

$$A_{1,1}X_1+A_{1,2}X_2+ \dots\dots +A_{1,n}X_n=A_{1,n+1}$$
$$A_{2,1}X_1+A_{2,2}X_2+ \dots\dots +A_{2,n}X_n=A_{2,n+1}$$
$$\dots\dots\dots\dots\dots\dots$$
$$A_{n,1}X_1+A_{n,2}X_2+ \dots\dots +A_{n,n}X_n=A_{n,n+1}$$

La prima volta si applica la seguente combinazione lineare a tutte le righe, esclusa la prima (K=1).

$$A_{i,j} = A_{i,j} - \frac{A_{K,j}}{A_{K,K}} \times A_{i,K}$$

Con i=1,2,....,n e i≠K
Con j=K+1,K+2,....,n+1
Poi si divide tutta la prima riga (K=1) per $A_{1,1}$ così il sistema diviene:

$$X_1+A_{1,2}X_2+ \dots\dots +A_{1,n}X_n=A_{1,n+1}$$
$$A_{2,2}X_2+ \dots\dots +A_{2,n}X_n=A_{2,n+1}$$
$$\dots\dots\dots\dots\dots\dots$$
$$A_{n,2}X_2+ \dots\dots +A_{n,n}X_n=A_{n,n+1}$$

La seconda volta (K=2) il procedimento si ripete, solo che invece di escludere la prima riga si esclude la seconda (la Kesima) e poi si divide tutta la seconda per $A_{2,2}$.

K poi diventa 3,4,....,n e quando il procedimento si è concluso (con K=n) il nostro sistema è diventato un sistema diagonale.

$$X_1 \qquad =A_{1,n+1}$$
$$X_2 \qquad =A_{2,n+1}$$
$$\dots\dots\dots\dots\dots\dots$$
$$X_n=A_{n,n+1}$$

le cui soluzioni sono ovviamente le stesse del

sistema iniziale. Il flow-chart è riportato in figura 1.
Per far girare il programma basta impostare il numero delle incognite del sistema (n) e premere B.
Impostare i valori della matrice per colonna partendo dalla prima cioè $A_{1,1}$ $A_{2,1}$ .... $A_{n,1}$ $A_{1,2}$ $A_{2,2}$ .... $A_{n,n+1}$
e premere A.
Alla fine premere E per dare inizio all'elaborazione, trascorso il tempo necessario il visualizzatore mostrerà X_1, premendo R/S si avrà X_2 e così via fino a X_n.
Aggiungiamo infine che per la TI-59 *n* può valere al massimo 9, mentre per la 58 *n* può arrivare a 5.

Fig. 1. Flow-chart del programma. Metodo di Gauss-Seidel per la soluzione di sistemi di n equazioni lineari in n incognite.

Analizzando il flow-chart (che è stato ridisegnato per chiarezza) si notano due blocchi fondamentali, oltre a quelli di semplice assegnazione, in cui compaiono operazioni su elementi di una matrice, che indicheremo genericamente con A_{rc}.

Vediamo ora i problemi che nascono quando si debba memorizzare una matrice e si richieda l'estrazione di un elemento: vedremo poi come vengono risolti.
Dobbiamo infatti ricordare che la matrice di partenza A è a «due dimensioni», cioè un singolo elemento è individuato da *due* coordinate *r* ed *s* (proprio come nella battaglia navale), mentre la memoria di cui disponiamo è ad «una dimensione», cioè possiamo accedere ad un elemento tramite un solo valore (ad es. con RCL 44). Il problema è dunque di «spezzare» opportunatamente la matrice in più parti, per poterla effettivamente memorizzare, avendo bene in mente che quando dovremo cercare un elemento *non* potremo più usare due coordinate ma *soltanto una*. Ecco che nasce dunque l'esigenza di avere una funzione a cui applicare i valori delle due coordinate e che dia come risultato un solo valore, che sarà proprio l'ubicazione dell'elemento voluto. Va da sé che questa funzione sarà differente a seconda del modo in cui «spezziamo» la matrice di partenza e risulterà molto semplice se operiamo una memorizzazione «per righe» o «per colonne». Nel nostro caso usiamo quest'ultimo metodo che consiste, come dice il nome, nel memorizzare, a partire da una certa cella di indirizzo I, prima tutta la prima colonna, poi la seconda e così via. Facendo riferimento alla fig. 2, dove sarà andato a finire l'elemento generico A_{rc} (indicato a tratteggio), cioè qual'è il suo effettivo indirizzo di memoria? Sappiamo solo che si trovava al punto di incrocio della riga *r* e della colonna c. Per trovarlo dobbiamo percorrere tutta la prima colonna, poi la seconda, fino alla (c - 1) - esima; poi nella colonna di indice *c* dobbiamo percorrere ancora (r - 1) elementi. In totale, dato che ogni colonna contiene *n* elementi, dobbiamo percorrerne (c - 1) n+r - 1, dopo di che siamo arrivati all'elemento desiderato. Perciò nella memoria, a partire dall'indirizzo I, dobbiamo arrivare all'indirizzo I_{rc} dato da:

$$I_{rc}=I+(c-1)\,n+r-1$$

È questa la funzione che cercavamo e dovrà essere applicata tutte le volte che nel programma si farà riferimento ad un elemento della matrice. Gli indirizzi degli elementi via via incontrati vengono memorizzati nei seguenti registri:

A_{ij} nella cella 05

A_{kj} nella cella 06

A_{jk} nella cella 07

A_{kk} nella cella 08

mentre viceversa i valori correnti di *i,j,k* sono rispettivamente nelle celle 01, 02, 03. Infine *n* viene memorizzato, come prima istruzione,

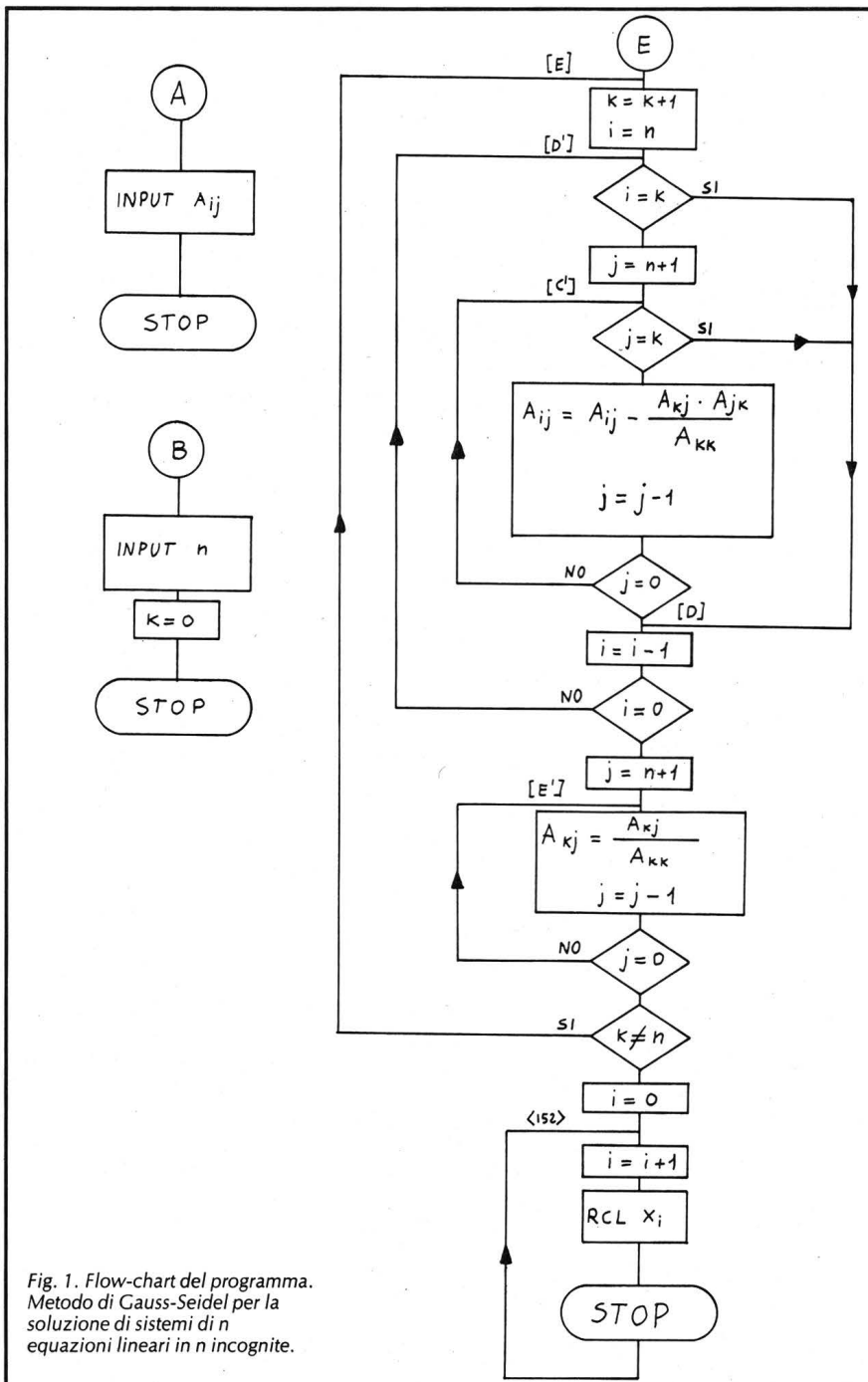
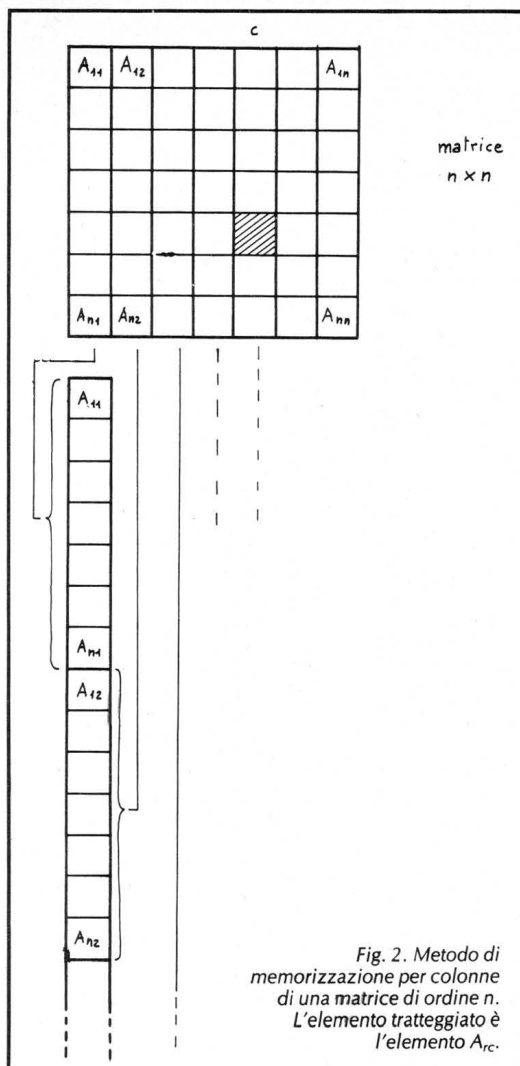
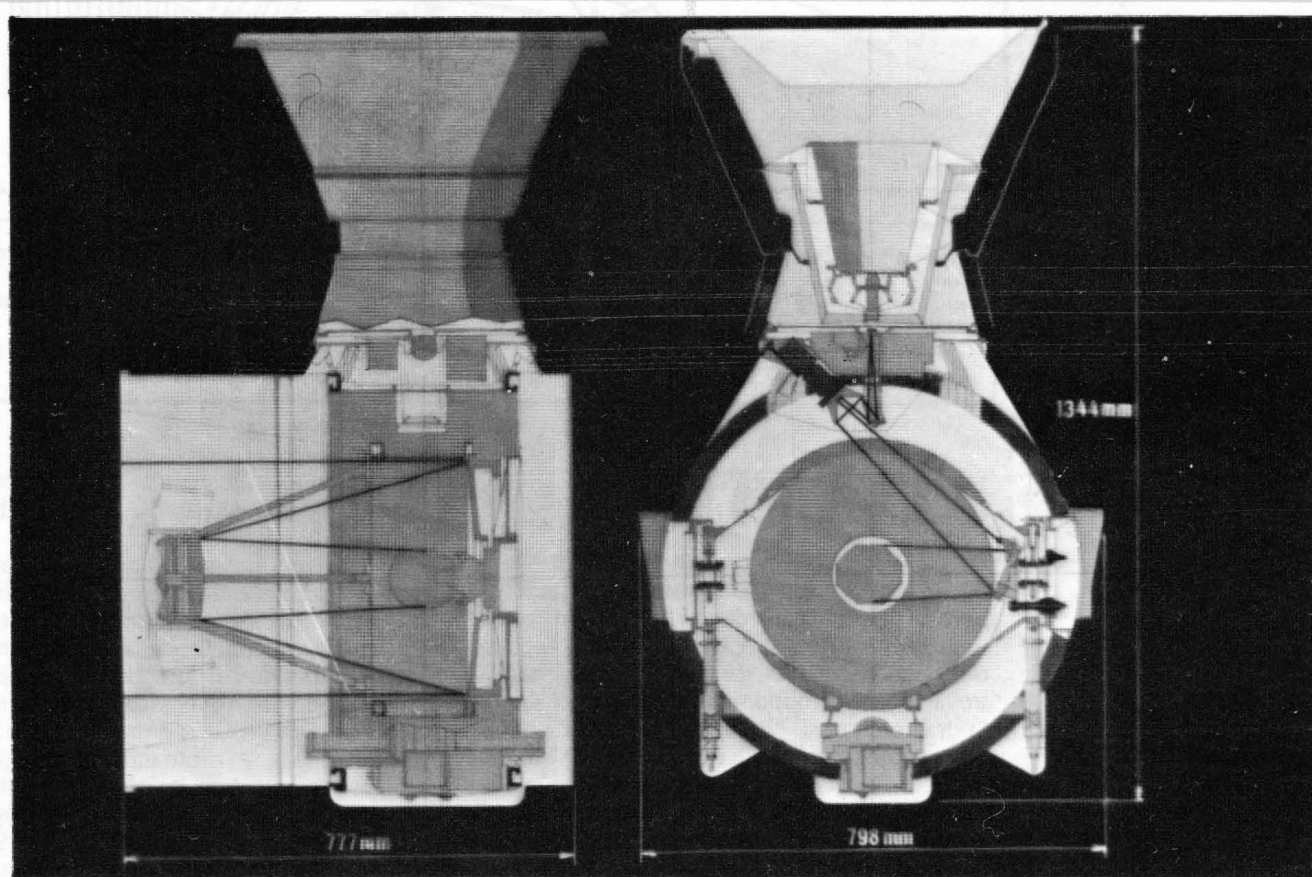



Fig. 2. Metodo di memorizzazione per colonne di una matrice di ordine n. L'elemento tratteggiato è l'elemento A_{rc}.

```
000  76 LBL    043  43 RCL    086  25 CLR    129  06  06
001  12  B     044  03  03    087  34 ΓX     130  73 RC*
002  42 STO    045  32 X:T    088  42 STO    131  06  06
003  00  00    046  43 RCL    089  06  06    132  55  ÷
004  00  0     047  00  00    090  43 RCL    133  73 RC*
005  42 STO    048  42 STO    091  03  03    134  08  08
006  03  03    049  01  01    092  71 SBR    135  95  =
007  09  9     050  76 LBL    093  33 X²     136  72 ST*
008  42 STO    051  19 D'     094  42 STO    137  06  06
009  09  09    052  43 RCL    095  05  05    138  97 DSZ
010  91 R/S    053  01  01    096  73 RC*    139  05  05
011  76 LBL    054  67 EQ     097  06  06    140  10 E'
012  11  A     055  14  D     098  75        141  43 RCL
013  69 OP     056  43 RCL    099  73 RC*    142  00  00
014  29  29    057  00  00    100  05  05    143  22 INV
015  72 ST*    058  42 STO    101  65  ×     144  67 EQ
016  09  09    059  02  02    102  73 RC*    145  15  E
017  91 R/S    060  69 OP     103  07  07    146  42 STO
018  76 LBL    061  22  22    104  55  ÷     147  05  05
019  33 X²     062  76 LBL    105  73 RC*    148  71 SBR
020  42 STO    063  18 C'     106  08  08    149  33 X²
021  04  04    064  43 RCL    107  95  =     150  42 STO
022  76 LBL    065  02  02    108  72 ST*    151  09  09
023  34 ΓX     066  67 EQ     109  06  06    152  69 OP
024  43 RCL    067  14  D     110  97 DSZ    153  29  29
025  05  05    068  43 RCL    111  02  02    154  73 RC*
026  65  ×     069  03  03    112  18 C'     155  09  09
027  43 RCL    070  42 STO    113  76 LBL    156  91 R/S
028  00  00    071  05  05    114  14  D     157  61 GTO
029  75  -     072  71 SBR    115  97 DSZ    158  01  01
030  43 RCL    073  33 X²     116  01  01    159  52  52
031  00  00    074  42 STO    117  19 D'
032  85  +     075  08  08    118  43 RCL
033  43 RCL    076  43 RCL    119  00  00
034  04  04    077  01  01    120  42 STO
035  85  +     078  71 SBR    121  05  05    001  12  B
036  09  9     079  33 X²     122  69 OP     012  11  A
037  95  =     080  42 STO    123  25  25    019  33 X²
038  92 RTN    081  07  07    124  76 LBL    023  34 ΓX
039  76 LBL    082  43 RCL    125  10 E'     040  15  E
040  15  E     083  02  02    126  71 SBR    051  19 D'
041  69 OP     084  42 STO    127  34 ΓX     062  18 C'
042  23  23    085  05  05    128  42 STO    114  14  D
                                             125  10 E'
```

Listing del programma per la soluzione di sistemi di equazioni lineari con il metodo di eliminazione diagonale.

nel registro 00; inoltre il flow-chart segue fedelmente l'andamento del programma, tanto che vengono pure indicate le etichette effettivamente usate.
Ora vogliamo fare delle considerazioni sull'algoritmo di Gauss-Seidel.
Nei blocchi «fondamentali» del flow-chart si hanno due frazioni con al denominatore lo stesso elemento A_{kk} (con k variabile).
Ora, in base ai dati iniziali o a seguito di un certo numero di calcoli, può capitare che uno degli A_{kk} risulti nullo, per cui le frazioni non sono calcolabili, facendo andare in errore il programma. A ciò, teoricamente, si può ovviare semplicemente scambiando l'equazione contenente il coefficiente A_{kk} incriminato con un'altra equazione tra quelle successive in modo tale che il nuovo A_{kk} sia diverso da zero. In pratica però si può evitare di arrivare all'«errore» per poi scambiare le equazioni, scegliendo *ogni volta*, tra le restanti equazioni, una di quelle aventi A_{kk} *maggiore* di una certa quantità prefissata; a questo punto si riordinano le equazioni e la relativa matrice, essendo certi che non si incorrerà più nell'errore di divisione. Ora tutto questo meccanismo avrebbe complicato parecchio il programma, allungandolo a dismisura e lasciando così poco spazio di memoria per i coefficienti: si veda ad esempio il Pgm 02 del modulo Master, il quale segue un procedimento che prevede il riordinamento delle equazioni (operazione di pivot parziale, come cita il manuale della biblioteca a pag. 13), ed è lungo ben 898 istruzioni! Nel nostro caso ci possiamo accontentare di far fermare il programma in caso di errore, appunto accendendo il flag n° 8 *prima di far partire l'elaborazione*. Tranquillizzatevi, noi abbiamo provato il programma per un sistema di 8° ordine, con coefficienti presi a caso: a parte la durata dell'elaborazione (per cui è vivamente consigliabile eliminare le etichette e passare ad indirizzamenti assoluti — è un utile esercizio!) non si sono verificati inconvenienti di sorta: anzi, i risultati sono *identici* a quelli ottenibili con il programma di biblioteca, che tra l'altro è nettamente più veloce. Un punto a favore per il programa di Andrianopulos, viene dal fatto che permette la risoluzione di sistemi del 9° ordine, cosa che il Pgm 02 non può fare a causa di una differente utilizzazione delle memorie. Saranno così felici coloro che, dovendo risolvere gravosi sistemi di 9 equazioni in 9 incognite, avevano già buttato fuori dalla finestra la calcolatrice, colpevole di non avere memorie a sufficienza... ■

COMPUTER COMPANY
AI CONFINI
DELL'IMPOSSIBILE
LE PIÙ ALTE PRESTAZIONI
AL PIÙ BASSO COSTO
Elaboratori e programmi dimensionati
secondo le esigenze personali del cliente.
Assistenza tecnica software-hardware
con interventi immediati.
Indiscutibile la competitività dei prezzi.
MIDIA NAPOLI
CERCANSI RIVENDITORI E CONCESSIONARI
PER ZONE LIBERE
COMPUTER
COMPANY
ELABORATORI ELETTRONICI
DIREZIONE GENERALE
VIA S.GIACOMO 32 · 80133 NAPOLI · TEL. (081) 310487 · 324786
SEDE DI ROMA
VIA MARIA ADELAIDE 4-6 · 00196 ROMA TEL. (06) 3611548 · 3606450

SOFTWARE R.P.N.

a cura di Paolo Galassetti

Inviate a m&p COMPUTER i vostri migliori programmi in R.P.N. (Notazione Polacca Inversa per calcolatrici Hewlett Packard).
I più interessanti, a giudizio della Redazione, saranno pubblicati sulla Rivista, e gli autori ricompensati con una biblioteca di programmi HP a loro scelta.
Si prega di inviare il materiale nella forma più ordinata possibile, scrivendo a macchina o comunque in maniera ben leggibile, e documentando i programmi con spiegazioni, commenti ed esempi.

FRAZIONE GENERATRICE DI UN DECIMALE QUALSIASI

```
01♦LBL "FRAZ"      33 STO 04
02 FIX 0           34 FRC
03 CLRG            35 X=0?
04 CF 29           36 GTO 02
05 CF 22           37♦LBL 06
06 "DECIMALE ?"    38 5
07 AVIEW           39 STO 02
08♦LBL 00          40 FS? 00
09 PSE             41 GTO 03
10 FC? 22          42 SF 00
11 GTO 00          43 GTO 04
12 STO 00          44♦LBL 02
13♦LBL 01          45 RCL 03
14 CF 00           46 RCL 02
15 1               47 /
16 ST+ 01          48 FRC
17 RCL 01          49 X≠0?
18 10↑X            50 GTO 06
19 STO 03          51 LASTX
20 RCL 00          52 STO 03
21 *               53 RCL 04
22 FRC             54 STO 00
23 X≠0?            55 GTO 04
24 GTO 01          56♦LBL 03
25 LASTX           57 CLA
26 STO 00          58 ARCL 00
27 2               59 "⊢/"
28 STO 02          60 ARCL 03
29♦LBL 04          61 AVIEW
30 RCL 00          62 STOP
31 RCL 02          63 END
32 /
```

Spesso, specialmente durante il corso di Analisi, mi sono trovato a dover eseguire lunghe e noiose differenze di decimali posti in forma frazionaria. In questi casi, si possono seguire due vie: o ci si mette di santa pazienza a trovare il MCD, oppure si computa ogni singola frazione e si opera sui decimali con il calcolatore. Il difetto di questo ultimo sistema, è che il risultato è un numeraccio con qualche cifra dopo la virgola, poco elegante (e poco accetto) per essere presentato, oltre che scomodo per eventuali calcoli successivi. Lo stesso problema, quando si trovano tramite programma, le radici di qualche equazione di secondo o terzo grado.

A questo punto si inserisce il mio programma, che altro non fa, a partire dal decimale, che fornire la frazione più semplice che gli corrisponde. Immettendo ad es. 0, 25, verrà fornita la risposta 1/4, da 9,12 si ricaverà 228/25 e così via. Premetto che la velocità di elaborazione è in media piuttosto elevata, pur dipendendo in una certa misura dal numero di cifre decimali (come risulterà chiaro dall'algoritmo).

Il principio su cui si basa il programma è semplice ma efficace: si trasforma il decimale in un intero moltiplicandolo per una appropriata potenza di dieci, e poi si opera sulla frazione ottenuta (decimale/potenza di dieci) in modo da semplificarla. Il tutto è preceduto da una breve routine di inserimento dati che accetta automaticamente la cifra impostata senza che sia necessario premere dopo né RUN/STOP né altri tasti.

Come preannunciato, le linee da due a undici, sono utilizzate, oltre che per la inizializzazione, anche per creare un anello tramite il flag 22, che si accende appena viene premuto un tasto numerico; il programma, dopo aver spento detto flag e aver visualizzato la richiesta di input (DECIMALE?) entra in anello interrogando il flag e ritornando (essendo questi disattivato) alla etichetta 00. Se a questo punto si batte una qualunque cifra, il programma attende, grazie alla pausa della linea 9, che il numero sia terminato, ed essendo a questo punto acceso il flag 22, esce dall'Anello e prosegue l'elaborazione.

Nelle linee che vanno dalla 12 alla 24, si provvede a moltiplicare il numero di ingresso per potenze crescenti di dieci fino a renderlo intero (FRC=0). Successivamente, si provvede a dividere numeratore e denominatore per 2 e/o per 5, cioè per gli unici numeri primi che dividono 10 e le sue potenze. Il programma ovviamente, inizia a dividere per 2 il numeratore e, se il risultato è un intero, effettua la stessa prova anche per il denominatore, memorizzando i nuovi risultati intermedi solo se anche quest'ultima prova è positiva. Dopo aver effettuato la stessa procedura anche per il 5, il puntatore salta, tramite il flag 00, all'etichetta di uscita 03, che tramite il registro ALFA e il simbolo / rende noto il risultato nella consueta forma NUM/DENOM.

Pierpaolo Bubbio - Torino

PIANO DI AMMORTAMENTO

Sono uno studente del liceo scientifico molto interessato all'informatica ed alla programmazione già da due anni. Ho trovato nella vostra rivista, un completo aggiornamento ed una soluzione per i problemi che la mia voglia di conoscere mi creava. Vorrei inoltre congratularmi con voi per la qualità della vostra rivista, ed in particolare sul fatto che con concorsi, richieste di programmi, e risposte ai problemi di coloro che vi scrivono, avete un vero e proprio dialogo con i lettori. Non c'è che dire: una rivista così ci voleva proprio.

Io sono in possesso di una calcolatrice programmabile HP-41C, definita giustamente «mostruosa macchinetta» da Antonio Finocchi (n. 4 di m&p COMPUTER), e, spinto dalle vostre sollecitazioni ad inviare i propri programmi, mi sono deciso a scrivervi.

Il programma è il piano di ammortamento. Questo è, nel campo della contabilità, uno dei più lunghi e faticosi calcoli. Il programma è veramente molto completo, ed è realizzabile con la sola memoria di base. Inutile dire che esso offrirà un valido supporto in questo campo, e renderà il tutto molto più pratico e veloce.

Tutti i dati necessari al programma vengono richiesti dallo stesso.

Prima bisogna dare il via al programma con XEQ «AMORT»,

alla domanda: «Cap.=?» immettere il debito,

alla domanda: «Tasso%=?» immettere il tasso di interesse,

alla domanda: «Anni=?» immetere il numero degli anni,

alla domanda: «Rate da mesi?» immetere il numero dei mesi di una rata (solo rate da mesi: 1 2 3 4 6 12),

alla domanda: «Vuoi la rata?» si può rispondere SI o NO, se si risponde SI, la calcolatrice calcola correttamente la rata annua di ammortamento, solo nel caso di rate da 12 mesi. Essa visualizza poi «Rata=n». Sfrutta la formula:

$$\text{RATA} = \frac{i\,((1+i)^N}{(1+i)^N-1} \text{ x}C.$$

Dove i=tasso%, N=anni, C=debito.

Se si risponde NO, la calcolatrice chiede: «Rata=?» immettere ora la rata scelta.

alla domanda: «Vuoi il D.R.?» la calcolatrice chiede se si vuol sapere direttamente il debito resudio all'ultima rata senza vedere tutto il piano di ammortamento. Si può rispondere SI o NO. Questa caratteristica serve per trovare la rata che si avvicini meglio ad un debito residuo uguale a zero. Naturalmente questo procedimento per tentativi è inutile se la rata è già stata calcolata dalla calcolatrice alla domanda precedente. Se si risponde NO, la calcolatrice inizia il vero e proprio piano di ammortamento.

Il piano di ammortamento, una volta iniziato, visualizza questa successione:

«Rata N. n» - Indica a *quale rata* è arrivato lo svolgimento del piano.

«Q.I.=n» - Indica la *quota interesse* della rata a cui è arrivata.

«Q.C.=n» - Indica la *quota capitale* della rata a cui è arrivata.

«D.E.=n» - Indica il *debito estinto* quando si arriva a questa rata.

«D.R.=n» - Indica il *debito residuo* quando si arriva a questa rata.

Alla fine dopo l'ultima rata verrà visualizzato:

«S.Q.I=n» - Indica la *somma delle quote interesse.*

«S.Q.C.=n» - Indica la *somma delle quote capitale* (che per prova deve essere uguale all'ultimo debito estinto visualizzato). Ognuno di questi risultati è accompagnato da un segnale acustico.

```
            XEQ "AMORT"
CAP.=?
    25,000,000.    RUN
TASSO%=?
            6.     RUN
ANNI?
            6.     RUN
RATE DA MESI?
            3.     RUN
VUOI LA RATA?
NO                 RUN
RATA?
     1,250,000.    RUN
VUOI IL D.R.?
NO                 RUN
RATA N.1.
Q.I.=375,000.
Q.C.=875,000.
D.E.=875,000.
D.R.=24,125,000.
RATA N.2.
Q.I.=361,875.
Q.C.=888,125.
D.E.=1,763,125.
D.R.=23,236,875.
RATA N.3.
Q.I.=348,553.
Q.C.=901,447.
D.E.=2,664,572.
D.R.=22,335,428.
RATA N.4.
Q.I.=335,031.
Q.C.=914,969.
D.E.=3,579,540.
D.R.=21,420,460.
RATA N.5.
Q.I.=321,307.
Q.C.=928,693.
D.E.=4,508,234.
D.R.=20,491,766.
RATA N.6.
Q.I.=307,376.
Q.C.=942,624.
D.E.=5,450,857.
D.R.=19,549,143.
RATA N.7.
Q.I.=293,237.
Q.C.=956,763.
D.E.=6,407,620.
D.R.=18,592,380.
RATA N.8.
Q.I.=278,886.
Q.C.=971,114.
D.E.=7,378,734.
D.R.=17,621,266.
RATA N.9.
Q.I.=264,319.
Q.C.=985,681.
D.E.=8,364,415.
D.R.=16,635,585.
RATA N.10.
Q.I.=249,534.
Q.C.=1,000,466.
D.E.=9,364,881.
D.R.=15,635,119.
RATA N.11.
Q.T.=234,527.
Q.C.=1,015,473.
D.E.=10,380,355.
D.R.=14,619,645.
RATA N.12.
Q.I.=219,295.
Q.C.=1,030,705.
D.E.=11,411,060.
D.R.=13,588,940.
RATA N.13.
Q.I.=203,834.
Q.C.=1,046,166.
D.E.=12,457,226.
D.R.=12,542,774.
RATA N.14.
Q.I.=188,142.
Q.C.=1,061,858.
D.E.=13,519,084.
D.R.=11,480,916.
RATA N.15.
Q.I.=172,214.
Q.C.=1,077,786.
D.E.=14,596,871.
D.R.=10,403,129.
RATA N.16.
Q.I.=156,047.
Q.C.=1,093,953.
D.E.=15,690,824.
D.R.=9,309,176.
RATA N.17.
Q.I.=139,638.
Q.C.=1,110,362.
D.E.=16,801,186.
D.R.=8,198,814.
RATA N.18.
Q.I.=122,982.
Q.C.=1,127,018.
D.E.=17,928,204.
D.R.=7,071,796.
RATA N.19.
Q.I.=106,077.
Q.C.=1,143,923.
D.E.=19,072,127.
D.R.=5,927,873.
RATA N.20.
Q.I.=88,918.
Q.C.=1,161,082.
D.E.=20,233,209.
D.R.=4,766,791.
RATA N.21.
Q.I.=71,502.
Q.C.=1,178,498.
D.E.=21,411,707.
D.R.=3,588,293.
RATA N.22.
Q.I.=53,824.
Q.C.=1,196,176.
D.E.=22,607,882.
D.R.=2,392,118.
RATA N.23.
Q.I.=35,882.
Q.C.=1,214,118.
D.E.=23,822,001.
D.R.=1,177,999.
RATA N.24.
Q.I.=17,670.
Q.C.=1,232,330.
D.E.=25,054,331.
D.R.=-54,331.
S.Q.I.=4,945,669.
S.Q.C.=25,054,331.
```

```
01•LBL "AMORT"
02 CLRG
03 "CAP.=?"
04 PROMPT
05 STO 00
06 "TASSO%=?"
07 PROMPT
08 STO 01
09 "ANNI?"
10 PROMPT
11 STO 02
12 "RATE DA MESI?"
13 PROMPT
14 STO 05
15 "VUOI LA RATA?"
16 AON
17 PROMPT
18 AOFF
19 ASTO 03
20 "SI"
21 ASTO 04
22 RCL 03
23 RCL 04
24 X≠Y?
25 GTO 00
26 100
27 ST/ 01
28 12
29 RCL 05
30 /
31 ST* 02
32 RCL 01
33 1
34 +
35 RCL 02
36 Y↑X
37 RCL 01
38 *
39 RCL 01
40 1
41 +
42 RCL 02
43 Y↑X
44 1
45 -
46 /
47 RCL 00
48 *
49 STO 03
50 "RATA="
51 ARCL 03
52 XEQ 13
53 SF 00
54•LBL 00
55 FS?C 00
56 GTO 09
57 "RATA?"
58 PROMPT
59 STO 03
60•LBL 05
61 "VUOI IL D.R.?"
62 AON
63 PROMPT
64 AOFF
65 ASTO 09
66 RCL 09
67 RCL 04
68 X=Y?
69 GTO 11
70 CF 01
71 GTO 08
72•LBL 11
73 SF 01
74•LBL 08
75 0
76 STO 04
77•LBL 07
78 1
79 ST+ 04
80 "RATA N."
81 ARCL 04
82 XEQ 13
83 RCL 00
84 RCL 01
85 %
86 XEQ IND 05
87 /
88•LBL 12
89 ST+ 06
90 "Q.I.="
91 ARCL X
92 XEQ 13
93 RCL 03
94 X<>Y
95 -
96 STO 08
97 ST+ 07
98 "Q.C.="
99 ARCL X
100 XEQ 13
101 "D.E.="
102 ARCL 07
103 XEQ 13
104 RCL 00
105 RCL 08
106 -
107 "D.R.="
108 ARCL X
109 XEQ 13
110 STO 00
111 RCL 04
112 RCL 05
113 *
114 12
115 /
116 RCL 02
117 X≠Y?
118 GTO 07
119 FS? 01
120 GTO 14
121 "S.Q.I.="
122 ARCL 06
123 XEQ 13
124 "S.Q.C.="
125 ARCL 07
126 XEQ 13
127 STOP
128•LBL 09
129 100
130 ST* 01
131 12
132 RCL 05
133 /
134 ST/ 02
135 GTO 05
136•LBL 01
137 12
138 RTN
139•LBL 02
140 6
141 RTN
142•LBL 03
143 4
144 RTN
145•LBL 04
146 3
147 RTN
148•LBL 06
149 2
150 RTN
151•LBL 13
152 FS? 01
153 RTN
154 AVIEW
155 BEEP
156 PSE
157 PSE
158 PSE
159 RTN
160•LBL 14
161 "D.R.="
162 ARCL 00
163 CF 01
164 AVIEW
165 BEEP
166 STOP
167 END
```

(da pag. 69)

Vi fornisco ora il listing del programma, e lo stato della calcolatrice prima di iniziarlo:
Size Ø1Ø
Fix Ø
Deg
(vedi pag. 69)

Il programma occupa 37Ø byte +1Ø registri dati.

Colgo l'occasione per porvi i più cordiali saluti.

Giuseppe Caggese
ROMA

Buona la forma usata per l'input-output dei dati, ma il tempo a disposizione per la lettura dei risultati è troppo breve (circa 3 secondi); inoltre, inserendo la stampante, tale pausa risulta superflua e ritarda inutilmente la stampa. Per eliminare tale inconveniente è sufficiente inserire, al posto degli XEQ 13, la funzione «AVIEW» e inserire all'inizio del programma (subito dopo LBL «AMORT») una istruzione SF 21; ovviamente così facendo la routine identificata con la LBL 13 (passi 151÷159) diventa inutile e va tolta. Conseguentemente alla modifica da noi proposta, non esisterebbe più la necessità di usare il flag Ø1 per inibire l'output dei dati indesiderati. Quest'ultimo compito può essere affidato al flag 21 sostituendo la sequenza:

Ø65 ASTO Ø9
....
....
....
Ø74 LBL Ø8

con la sequenza:

Ø65 ASTO X
Ø66 RCL Ø4
Ø67 X=Y3
Ø68 CF 21

e sostituendo i passi

163 CF Ø1 con SF 21
119 FS? Ø1 con FC? 21

In tal modo si hanno quattro possibilità, a seconda della risposta data dalla domanda «Vuoi il D.R.?», e a seconda che sia collegata o no la stampante:
1) risposta «SI» senza stampante: la macchina visualizza rapidamente tutti i risultati intermedi, senza arrestarsi, e si ferma solo dopo aver calcolato e visualizzato il risultato finale (D.R.).
2) Risposta «NO» senza stampante: la macchina si ferma ad ogni risultato, per procedere si preme il tasto R/S.
3) Risposta «SI» con stampante: vengono stampati tutti i risultati.
4) Risposta «NO» con stampante: viene stampato soltanto il risultato finale (D.R.).

CALCOLO FRAZIONARIO

Vi invio un breve programma che permette di effettuare le 4 operazioni con numeri frazionari. Esso non usa alcun registro di memoria dati ma utilizza soltanto la catasta e Lastx.
I risultati vengono ridotti ai minimi termini tramite la routine Ø1 che calcola il massimo comun divisore di due numeri contenuti in X e Y. Se i due operandi sono A/B e C/D i dati vanno impostati secondo la sequenza A ENTER B ENTER C ENTER D e i risultati vengono visualizzati nella forma E/F. Nel «listing» (si fa per dire) purtroppo scritto a mano (per carenza cronica di stampanti) che allego, i simboli da me utilizzati sono simili a quelli della anelata stampante con la sola eccezione del simbolo di elevamento a potenza e al quadrato, per i quali ho utilizzato le notazioni consuete.
Complimentandomi per l'ottima rivista spero che questo programma che a me (studente di fisica del primo anno) risulta utile, appaia altrettanto interessante a voi.

Bagliesi Giuseppe,
MONTELEPRE (PA)

Il programma del Signor Bagliesi, che abbiamo fatto listare dalla nostra stampante, è senz'altro assai utile e pratico da usarsi; cone le assegnazioni da lui proposte, la macchina, nel modo «USER», fa corrispondere ai tasti +, –, x e ÷ le quattro operazioni sulle frazioni.
OK al Signor Bagliesi: per lui la catasta non ha segreti!

```
01♦LBL "/"
02 X<>Y
03♦LBL "*"
04 X<> T
05 *
06 GTO 00
07♦LBL "-"
08 CHS
09♦LBL "+"
10 X<> Z
11 *
12 LASTX
13 RDN
14 RDN
15 *
16 LASTX
17 X<> T
18 +
19♦LBL 00
20 X<> Z
21 *
22 X<>Y
23 STO T
24 X<>Y
25 STO Z
26♦LBL 01
27 MOD
28 X=0?
29 GTO 02
30 LASTX
31 X<>Y
32 GTO 01
33♦LBL 02
34 X<> L
35 ST/ Z
36 ST/ Y
37 RDN
38 CLA
39 ARCL Y
40 "├/"
41 ARCL X
42 AVIEW
43 END
```

il microcomputer nasce dal bus

Parte quarta

Dalla CPU al sistema di sviluppo - IL LIVELLO 2

Lorenzo Mezzalira

Un modulo CPU organizzato come single board computer è già in grado di supportare interessanti esperienze o semplici applicazioni.
Per diventare un sistema di sviluppo che permetta la realizzazione e la messa a punto di progetti hardware e software occorre naturalmente arricchire la dotazione di risorse, a corredo della CPU, in termini di memoria, software di sistema, periferiche e relative interfacce.
In questo articolo ci si propone di esaminare le caratteristiche di un semplice sistema di sviluppo e di vedere come esso possa nascere e crescere gradualmente con l'impiego di mobili della serie MMS-8, realizzati da L'EMMECI e compatibili con il bus della CPU-03, presentata come LIVELLO 1 nella parte terza di questa serie, e, come l'adozione di un MONITOR modulare permetta all'utente di personalizzare il sistema.

Caratteristiche di un sistema di sviluppo

Un sistema di sviluppo è costituito da un calcolatore che serve come strumento di ausilio ai tecnici per le seguenti attività:
— realizzazione dei programmi
— messa a punto (debug) dei programmi
— collaudo e messa a punto di moduli.
Ognuna di queste attività ha una sua particolare problematica e può essere svolta con un maggiore o minore livello di sofisticazione in base alla disponibilità o meno di particolari strumenti hardware o software, oltre che, naturalmente, alla capacità di servirsene da parte degli utilizzatori.
Per meglio comprendere la natura e l'uso degli strumenti, è utile una breve presentazione delle singole attività.

1. Realizzazione dei programmi

È la fase che parte dalla fine del progetto di un programma (schema a blocchi e specifiche sulle strutture dati) e giunge alla disponibilità su un supporto di memoria del programma eseguibile, cioè, come si suol dire, in «codice macchina».
Le attività connesse con questa fase consistono essenzialmente nell'introduzione del programma «sorgente» codificato in una forma opportuna (linguaggio), nell'eventuale modifica (editing) del programma sorgente e nella sua conversione in codice macchina.
Le prestazioni minime consistono quindi nella possibilità di introdurre in formato esadecimale le istruzioni che verranno direttamente memorizzate in forma eseguibile, e nella possibilità di modificare il contenuto di singole celle di memoria.
Prestazioni maggiori e, ovviamente sempre più interessanti, si ottengono quanto più il formato di ingresso, cioè il linguaggio sorgente è vicino al modo di pensare del preogettista e svincolato dalla particolarità della macchina. In questi casi, la fase di conversione in formato eseguibile è una vera e propria traduzione che prende il nome di assemblaggio (per il linguaggio assembler) o di compilazione (per i linguaggi ad alto livello come FORTRAN, PASCAL, ecc.).
Il massimo della potenzialità si ottiene quando sia possibile gestire biblioteche di programmi e di definizioni che possono facilmente venire richiamati e collegati tra loro.

2. Messa a punto dei programmi

Questa fase, detta di debug, è logicamente seguente alla realizzazione dei programmi, anche se nell'effettiva attività del programmatore si ha una continua alternanza tra le due fasi, cioè quella di modifica e quella di prova.

TERMINOLOGIA

Editor: programma che permette di modificare un testo (per esempio un programma sorgente) mediante inserimenti, sostituzione o cancellazione di caratteri, parole o linee. È indipendente dal tipo di microprocessore e di linguaggio usati.

Linguaggio Assembler: consente la formulazione mediante sigle mnemoniche delle istruzioni macchina: è dipendente dal microprocessore.

Programma Assembler: esegue la traduzione di ogni istruzione in linguaggio Assembler nella corrispondente forma in codice macchina.
È dipendente dal microprocessore.

Linguaggi ad alto livello (High Level Languages): consentono la formulazione di programmi in modo indipendente dal particolare microprocessore e più vicino alle esigenze del programmatore.
Compilazione: traduzione di un programma «sorgente» scritto in un linguaggio ad alto livello per produrre il programma «oggetto» in codice macchina, che potrà essere direttamente caricato in memoria ed eseguito. Il programma sorgente è indipendente dalla macchina mentre non lo è il programma oggetto.

Interprete: programma di sistema che «leggendo» un programma applicativo presente in memoria, scritto in un codice diverso dal codice macchina, lo «interpreta» eseguendo le operazioni che il programma applicativo rappresenta.
L'interprete dipende dalla macchina su cui gira, mentre il programma applicativo può essere scritto in un linguaggio indipendente dalla macchina (es.: BASIC, P-code, ecc.).

Bootstrap: programma di sistema residente su EPROM che prende il controllo all'accensione della macchina e ne inizializza il funzionamento, eventualmente caricando opportuni programmi di sistema dalla memoria di massa.

Loader: programma di sistema che carica in memoria i programmi in codice macchina, per la loro esecuzione.

Rilocabile: programma oggetto non direttamente eseguibile perché non ancora assegnato ad una precisa area di indirizzamento.

Linking-loader: programma di sistema che carica in una data area di indirizzamento moduli rilocabili rendendoli eseguibili.

Fig. 1. Configurazione minima di sistema di sviluppo LIVELLO 2, adatta per lavorare in codice esadecimale.

Il debug sostanzialmente consiste nell'individuare e correggere il maggior numero possibile degli inevitabili errori di programmazione, e nella taratura degli eventuali parametri di personalizzazione.
Ciò è generalmente ottenuto fornendo opportuni dati in ingresso ed osservando attentamente il precedere dell'esecuzione del programma e i risultati che esso fornisce.
I requisiti minimi consistono nel poter «lanciare» l'esecuzione di un programma a partire da un indirizzo precisato e con valori prefissabili dei dati in ingresso, nel poter arrestare l'esecuzione ad un indirizzo prefissato e osservare lo stato della macchina (registri della CPU e memoria) al momento dell'arresto (breakpoint).
Prestazioni più sofisticate permettono una maggiore varietà di funzioni, eventualmente correlate con il linguaggio sorgente invece che con il più macchinoso codice macchina, e una maggior flessibilità di visualizzazione della situazione interna, che faciliti la «diagnosi» da parte dell'operatore.

3. Collaudo e messa a punto di moduli

Il collaudo di parti di un microcalcolatore può essere vantaggiosamente effettuato generando i segnali che stimolino le varie funzionalità dell'elemento da collaudare ed osservando le sue reazioni e risposte.
Questa osservazione potrà essere effettuata tramite strumenti esterni (oscilloscopi e analizzatori di stati), ma spesso è più comodo che sia ottenuta tramite una interpretazione del comportamento del modulo in esame da parte dello stesso sistema di sviluppo che ha generato gli stimoli. Come è evidente, la casistica e le possibilità di personalizzazione di queste operazioni sono molto vaste.
Un'altra attività comoda, se non indispensabile, consiste nella possibilità di programmare direttamente EPROM e PROM che molto spesso sono utilizzate per personalizzare un modulo con la realizzazione di opportune funzioni combinatorie.

Aspetti hardware - sistema modulare

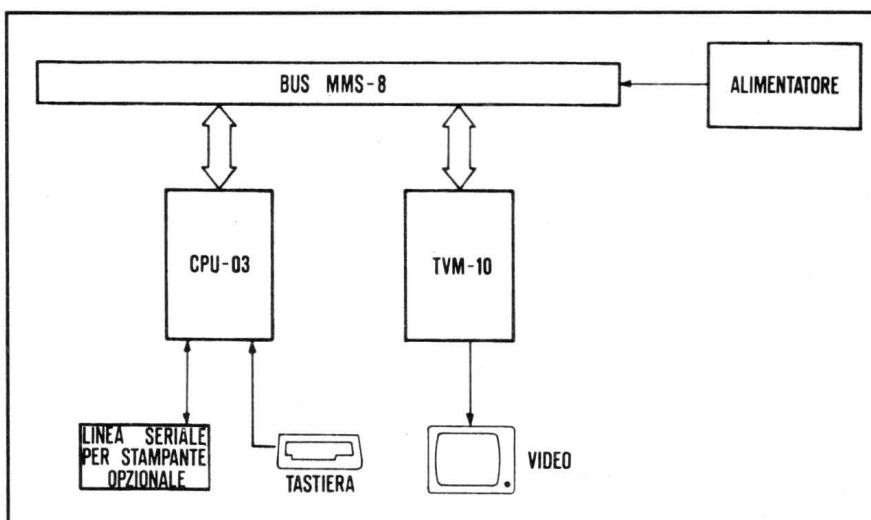
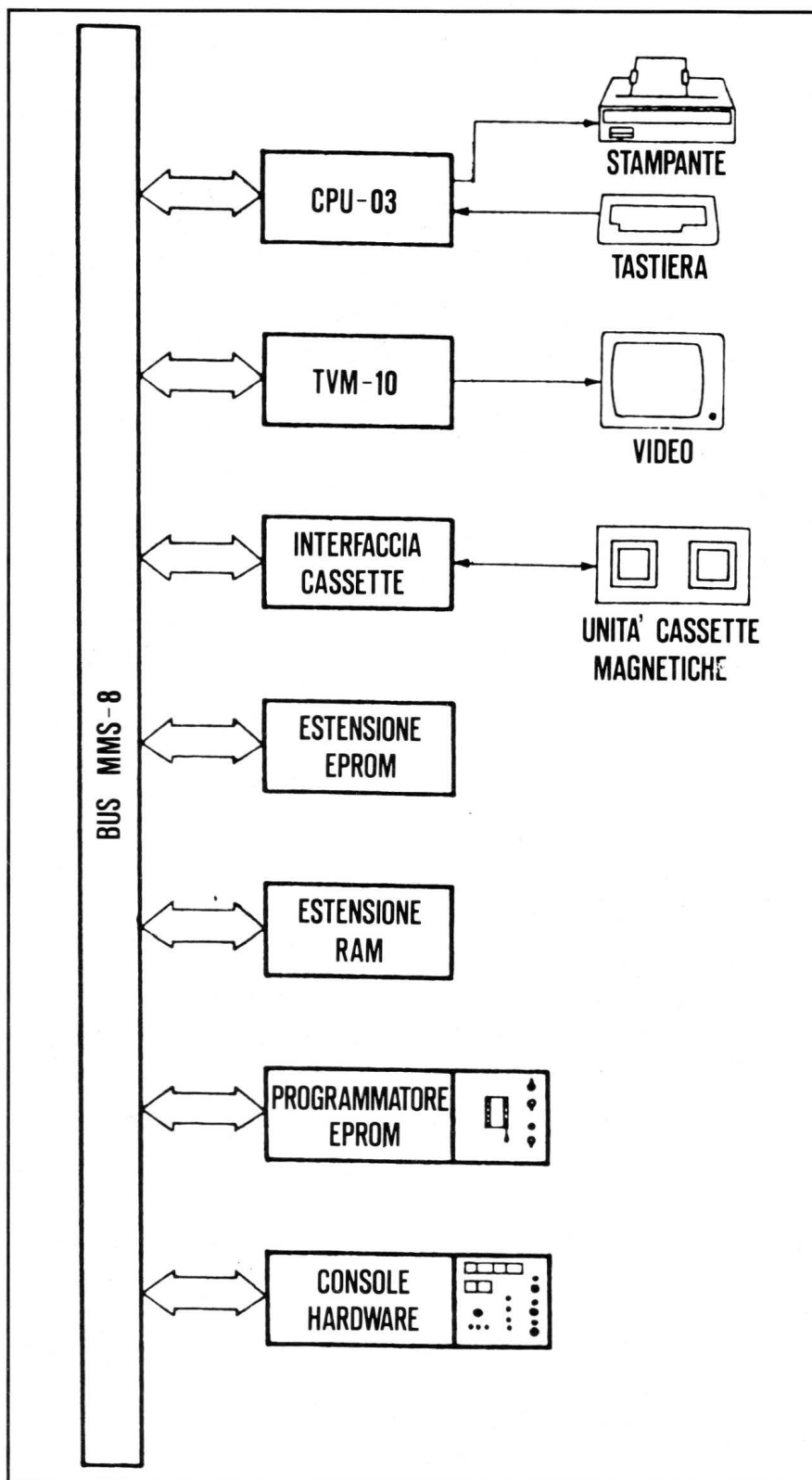
Vediamo ora quale deve essere la configurazione circuitale per poter supportare le funzioni a cui si è accennato nel paragrafo precedente.
È però utile, come introduzione a questo paragrafo, classificare le soluzioni che si possono adottare a livello hardware.
1. Impiego di un calcolatore «general purpose» non basato sul microprocessore con cui verrà realizzata l'applicazione (HP 64000, Tektronix 8001).
2. Impiego di un particolare microcalcolatore basato sullo stesso microprocessore usato in applicazione, ma con struttura circuitale diversa ed orientata allo sviluppo (Intel MDS, Zilog, ecc.).
3. Configurazione modulare del sistema di sviluppo, utilizzando gli stessi moduli della realizzazione applicativa, arricchita delle risorse utili in sede di sviluppo.
Quest'ultima soluzione è stata adottata da l'Emmeci s.r.l. e collaudata da quasi un quinquennio di lavoro.

Senza voler approfondire il confronto tra le caratteristiche degli approcci citati, si possono riassumere gli aspetti interessanti della filosofia modulare con le considerazioni seguenti:
— un sistema modulare può crescere con le esigenze;
— l'impiego degli stessi moduli per lo sviluppo e l'applicazione rende più reale e concreto il collaudo e facilita molto le modifiche e manutenzione sull'applicazione realizzata;
— l'operatore non deve tener presenti due contesti diversi (quello di sviluppo e quello applicativo), ma la sua realizzazione nasce e si consolida nello stesso sistema che egli ben conosce.
In conclusione, la disponibilità di opportuni moduli, oltre che del software di cui si parlerà nel prossimo paragrafo, permette la realizzazione di efficaci sistemi, il cui costo è facilmente proporzionale alle prestazioni richieste.
Partendo da un modulo microcalcolatore su singola scheda, come la CPU-03, le risorse aggiuntive necessarie sono:
— contenitore per le schede, corredato di connettori collegati a Bus
— alimentatore
— terminale interattivo per l'operatore.
Quest'ultimo sarà composto da una tastiera, collegabile direttamente ad una porta di ingresso della CPU, e da uno schermo video che richiede un modulo di interfaccia, come la TVM-10, che contiene la memoria e la circuiteria necessaria per il rinfresco dell'immagine e la generazione dei caratteri alfanumerici.
Questa configurazione minima, rappresentata nella Fig. 1, realizza già interessanti funzioni, ma è ancora carente su alcuni punti spesso essenziali.
In particolare è tipico che si voglia giungere alla configurazione illustrata nella Fig. 2, ottenuta aggiungendo le seguenti risorse:
— espansione memoria RAM, utile sia per programmi applicativi più estesi, che per l'adozione di software di sivluppo (per esempio Assembler)
— espansione memoria EPROM che arricchisce la disponibilità di software di utilità
— cassette magnetiche che costituiscono la forma più economica di memoria di massa per contenere sia i programmi sorgenti che quelli direttamente caricabili ed eseguibili
— stampante per fornire copie permanenti dei «listing» dei programmi
— modulo programmatore di EPROM per trasferire su memorie non volatili i programmi e i parametri che abbiano una forma definitiva e debbano essere resi sempre disponibili
— modulo «console hardware» che permette l'esecuzione «single-step» e «single-instruction» di programmi, visualizzando direttamente indirizzi, dati e segnali di controllo sul Bus.
Questa classe di configurazioni, che chiameremo «Livello 2», permette lo sviluppo di programmi scritti in esadecimale o in Assembler, a seconda della configurazione, e il loro debug.
Un'altra possibilità è quella di adottare un interprete BASIC: in questo caso è tipico che sistema di sviluppo e sistema applicativo vengano esattamente a coincidere.

Fig. 2. Configurazione tipica del LIVELLO 2, adatta per lavorare in Assembler e/o in BASIC.

Aspetti software - Monitor

Le risorse hardware, peraltro necessarie, assumono una funzionalità utile solo quando siano presenti i programmi che «danno vita» al calcolatore. Non è esagerato affermare che un buon software è almeno altrettanto (se non più) importante di un buon hardware.
Ecco quindi che una certa configurazione modulare assume il ruolo di sistema di sviluppo solo se corredata da un insieme di programmi, il cui nucleo risiede su EPROM, che offrono le prestazioni necessarie e permettono la gestione delle unità periferiche di vario tipo.

Il nucleo base, che può essere considerato un semplice sistema operativo, è detto comunemente «Monitor» e deriva la sua importanza dal fatto che costituisce appunto la base su cui si appoggiano tutti gli altri programmi di sistema e di utilità.
Questa importanza del Monitor suggerisce che esso, venga accuratamente progettato, realizzato ed aggiornato.
È immediato notare che un Monitor che debba gestire sistemi modulari in configurazioni anche molto varie deve essere facilmente adattabile ed espandibile, cioè presentare esso stesso un'organizzazione modulare.
Prendiamo ad esempio il Monitor MON-80 realizzato da L'EMMECI per i suoi sistemi di sviluppo dal Livello 2 in sù.
Questo Monitor presenta due direzioni di espansione modulare:
— comandi dell'operatore
— gestione delle unità periferiche.
Per quanto riguarda i comandi dell'operatore, l'organizzazione è la seguente. Ogni comando è realizzato mediante un programma, che costituisce un modulo software, corredato di un identificatore di tipo che lo qualifica appunto come comando e di un identificatore simbolico (cioè un nome) che ne permette l'individuazione univoca.
Ognuno di questi programmi, che possono anche essere scritti dall'utente ad integrazione del corredo base, può utilizzare funzioni di ingresso/uscita e subroutine di utilità messe a disposizione dal nucleo mediante punti di ingresso (entry point) collocati in una tabella ad indirizzi noti.
Il nucleo base, con i caratteri digitati dall'operatore sulla tastiera, compila una stringa che generalmente avrà il fomato:

<nome> <terminatore> [<lista parametri>] <CR>

Il «nome», che identifica il comando, è utilizzato per una ricerca in memoria dell'intestazione del programma corrispondente. Se questo non viene trovato, viene segnalata la sua inesistenza con la presentazione su video di un «?»; se invece il programma esiste, gli vengono passati gli eventuali parametri e gli viene ceduto il controllo che, al termine delle operazioni tornerà al Monitor per ripetere le operazioni descritte con altri comandi dell'operatore.
In questo modo l'aggiunta di nuovi comandi non comporta nessuna modifica della parte già esistente, mentre l'invocazione di comandi non implementati fornisce una semplice segnalazione diagnostica.
L'organizzazione dell'ingresso/uscita è invece la seguente.

Sono previste cinque periferiche «logiche» cui fanno riferimento tutti i programmi, che hanno il significato di:
1) Ingresso da console operatore (CI)
2) Ingresso da lettore di dati (RI)
3) Uscita su console operatore (CO)
4) Uscita per memorizzazione esterna (PO)
5) Uscita per stampa (LO).
Per ognuna di queste periferiche logiche, è previsto un certo numero di operazioni logiche come: inizializzazione, trasferimento di un carattere, chiusura, ecc., che potranno essere invocate dal programma chiamante.
Esiste poi una collezione di «handler» che sono le routine di gestione fisica delle varie periferiche. Per ogni periferica e relativa interfaccia è necessario un diverso handler che tenga conto delle effettive modalità di gestione proprie di quella periferica e che realizzi nel modo più opportuno le funzioni logiche prima citate.
Con un opportuno comando è possibile «agganciare» ed inizializzare automaticamente una qualsiasi periferica fisica con una qualsiasi periferica logica, con il solo ovvio vincolo di compatibilità che esse siano entrambe in ingresso o in uscita. Ciò è ottenuto, da parte dell'operatore, invocando il nome del comando e fornendo come parametri il nome della periferica logica e il nome dell'handler. Naturalmente sta al buon senso dell'operatore agganciare alle funzioni logiche periferiche adatte allo scopo - un nastro magnetico non si presta, per esempio, come uscita su console.
Anche per gli handler, si ha una ricerca in memoria dell'intestazione con il relativo identificatore: ciò rende estremamente facile l'aggiunta di nuovi handler e rapida la commutazione su diverse periferiche fisiche delle funzioni logiche.
Oltre al Monitor, corredato dei comandi più utili e degli handler delle periferiche usate, se si dispone di unità a cassette magnetiche è generalmente prevista la dotazione di un Editor per introdurre e modificare i programmi sorgenti e un Assembler per tradurli in codice macchina. Questa costituisce la configurazione software tipica per lo sviluppo di programmi applicativi del tipo «controllo in tempo reale» e gestione di ingresso/uscita.
Con la stessa configurazione hardware si può avere una configurazione software di impronta diversa utilizzando un interprete BASIC.
Questo linguaggio ad alto livello è solitamente utilizzato in modo «interattivo», il che significa che l'operatore colloquia con l'interprete che svolge, sia pure embrionalmente, le funzioni di Editing, di traduzione (durante l'esecuzione del programma) e di debug.
Usualmente, almeno per l'I/O, l'interprete si appoggia al Monitor, ma per il resto costituisce un sistema software autosufficiente.
Come si è detto, in questo caso il sistema di sviluppo coincide con il sistema applicativo dato che produce programmi destinati ad essere eseguiti da sé stesso e in genere non su altre macchine.
Il linguaggio BASIC è adatto per applicazioni scientifiche (calcolo) e gestionali in cui non siano richieste particolari prestazioni di velocità.

L. Mezzalira, R. Bisiani, R. Negrini - Una proposta di bus standard per microcalcolatori modulari - Elettronica Oggi - Febbraio 1979
L. Mezzalira - Il microcomputer nasce dal bus. Introduzione - Micro & Personal COMPUTER - N. 4
L. Mezzalira - Il micorcomputer nasce dal bus. Progetto di un modulo CPU con microprocessore. - Micro & Personal COMPUTER - N. 5
L. Mezzalira - Il microcomputer nasce dal bus. Dall'8080 al Single Board Computer sul Bus MMS-8 - Micro & Personal COMPUTER - N. 7
P. Colombo - Il componente multifunzione TMS-5501 - Informazione Elettronica - Febbraio 1979

Via Bartolomeo della Gatta 26/28 - tel 055/713369 - 50143 Firenze

SOFTWARE PER GESTIONE AZIENDALE

GESTIONE MAGAZZINO L. 360.000 + IVA
Un programma professionale per il Vs. Apple. La riduzione delle scorte e la statistica sulla movimentazione resa possibile da una efficiente gestione del magazzino, ripaga largamente da sola l'acquisto di un Apple. Richiede 48K di memoria, da 1 a 4 dischi, eventuale stampante.

FATTURAZIONE E BOLLETTAZIONE L. 360.000 + IVA
Automatizzate l'emissione di bolle e fatture con Apple. Il Programma attinge informazioni dall'archivio Clienti e dall'archivio di magazzino ed è collegato alla successiva contabilità IVA. Richiede 48K di memoria da 1 a 4 dischi, Stampante.

CONTABILITÀ CLIENTI L. 420.000 + IVA
Risolvete finalmente con Apple il problema di una gestione efficiente delle Vendite. Automaticamente estratti conto, giornali bollati IVA, controllo del Credito, analisi del Fatturato, etc.

CONTABILITÀ FORNITORI L. 420.000 + IVA
Forse più delle vendite Vi preme gestire bene i Vs. acquisti. Le stesse prestazioni della contabilità Clienti con 2 giornali IVA. (disponibile da giugno 80)

CONTABILITÀ GENERALE L. 420.000 + IVA
L'ovvio completamento di una contabilità sezionale unito ai precedenti. Utilizzato da solo, se non avete problemi di IVA, Vi fornirà sempre un'aggiornata situazione del piano dei conti e, cosa più importante, un giornale fiscale e la situazione del conto "profitti e perdite". (disponibile da Maggio 80)

AUTOLISTINO L. 280.000 + IVA
Un listino-Prezzi Intelligente destinato ad imprese commerciali di vendita all'ingrosso e al dettaglio. Attinge informazioni dall'archivio magazzino, incrementa o decrementa il prezzo di gruppi di articoli o intere categorie. Ricava il prezzo vendita in base ad un'aliquota prefissata. Calcola automaticamente gli sconti per quantitativi.

PAGHE E STIPENDI L. 360.000 + IVA
Un programma professionale e generalizzato che soddisfa le esigenze di 15 diversi contratti Nazionali di Lavoro.
Particolarmente interessante per i Consulenti.

PAGHE EDILI L. 360.000 + IVA
Una versione speciale del programma PAGHE E STIPENDI per gli operai del settore Edilizio.

DISTRIBUZIONE GIORNALI L. 500.000 + IVA
Risolvere tutti i problemi di gestione delle Agenzie di distribuzione Giornali alle Edicole, compresa la gestione delle spedizioni, gestione resi, emissione di bolle e fatture.
Prevede fino a 600 Testate in distribuzione fino a 125 Edicole.

ANALMEDIC 1 L.450.000 + IVA
Programma per Laboratori di Analisi Mediche: Gestione completa del Libro di ingressi o ACCETTAZIONE. Prepara automaticamente il giornale delle analisi da effettuare dopo i prelievi giornalieri, stampa le etichette per le provette, e i moduli e/o veline dei risultati (per l'archivio) come da Normativa Vigente.
È composto da un menù principale e da 12 menù secondari, suddivisi per tipo di analisi.
Richiede 48K, 2 dischi, Stampante.

ANALMEDIC 2 L. 320.000 + IVA
Programma per Laboratori di Analisi Mediche: Gestione completa dell'archivio clienti, registra tutte le analisi fatte e loro data, stampa la scheda dell'analisi con i risultati, è legato al LIBRO INGRESSI del precedente ANALMEDIC 1.
Richede 48K, 2 dischi, Stampante.

AMMINISTRAZIONE CONDOMINI L. 360.000 + IVA
Programma completo per la gestione condominiale, calcola automaticamente le quote suddivise in millesimi voce per voce, gestisce la posizione dei condomini per acconti, saldi, sospesi. Effettua la stampa dettagliata del giornale di gestione e degli estratti conto.
Richiede 48K, 1 o 2 dischi, Stampante.

COMMERCIO E DISTRIBUZIONE 1 L. 450.000 + IVA
Un Programma destinato ad imprese commerciali all'ingrosso e/o dettaglio: attinge i dati dall'archivio creato col programma MAGAZZINO, da la possibilità dell'articolo/i richiesto/i, calcola l'IVA, stampa un giornale delle movimentazioni effettuate (numerate conindice azzerabile e/o impostabile), crea un archivio movimentazioni, effettua la statistica di tutte le operazioni eseguite, scarica automaticamente le merci vendute dall'archivio MAGAZZINO.
Richiede 48K, almeno 2 dischi, Stampante.

COMMERCIO E DISTRIBUZIONE 2 L.450.000 + IVA
Simile al precedente, con lo scarico e la stampa dettagliata suddivisa in più reparti dell'azienda, analoga suddivisione per la parte statistica.
Richiede 48K, almeno 2 dischi. Stampante.

GESTIONE ALBERGHIERA COMPUTERZZATA

FRONT OFFICE PER ALBERGHI L. 400.000 + IVA
La soluzione ideale per Alberghi di qualsiasi dimensione, dalle 50 camere in sù. Gestione delle Prenotazioni nel tempo (fino ad un anno). Gestione degli arrivi con quadratura contabile con la saldaconti. Gestione statistica (Presenze) ed economica (media per presenze). Gestione vendita Saloni.

BACK OFFICE PER ALBERGHI (in preparazione)
AMMINISTRAZIONE PER ALBERGHI (in preparazione)
GESTIONE RISTORANTE, BAR, MAGAZZINO (in preparazione)

VISICALC L. 175.000 + IVA
Un potente ed originalissimo strumento di archiviazione e di calcolo. Trasforma Apple in una grande matrice di dati legati fra loro da relazioni semplici o complesse.
Cambiando un dato vengono automaticamente ricalcolati quelli che da esso dipendono.
Ideale per la computazione di calcoli contributivi, scala mobile, Listini prezzi con molte scale sconti, etc.
Richiede 48K, 1 disco, Stampante.

ALLESTIMENTO E GESTIONE ARCHIVI

DATA BASE MANAGEMENT SYSTEM 1 L. 250.000 + IVA
Uno strumento di grande generalità. Consente di sviluppare immediatamente la maggior parte delle piccole e medie applicazioni gestionali di Apple, senza scrivere una riga di Programmi.
Fra le caratteristiche principali: introduzione, aggiornamento di dati, ordinamenti (fino a 20) anche multichiave, stampe personalizzate, ricerche selettive, totalizzazioni. Richiede 48K di memoria, 2 dischi, Stampante.

MINI DATA BASE L. 110.000 + IVA
Una versione ridotta del Programma precedente. Richiede 48K di memoria, 1 disco, eventuale Stampante.

DATA BASE MANAGEMENT SYSTEM 2 L. 360.000 + IVA
Una versione più estesa del System 1, ideale per usi anagrafici, Statistici, Pubblicitari indirizzati a specifici settori merceologici, Ordinamenti e ricerche per Laboratori di analisi Mediche, ed altre vastissime applicazioni. Richiede 48K di memoria, almeno 2 dischi, Stampante.

WHATSIT - DATA BASE A CAMPI LIBERI L. 175.000 + IVA
Un originalissimo archivio a campi liberi, si possono cioè diminuire od aumentare i dati di un nominativo senza essere legati ad uno schema predeterminato.
Si interroga la Macchina come interrogare una Persona: es. "quando l'appuntamento con l'ing. ROSSI?" - risposta "l'appuntamento con l'ing. ROSSI è alle 10 del 23/5/80
Il programma è in lingua inglese, ma i dati possono essere inseriti in Italiano. Richiede 48K di memoria. 1 disco. linguaggio Integer Basic.

ELABORAZIONE DI TESTI - WORD PROCESSING -

APPLE WRITER L. 79.000 + IVA
Trasformate il Vostro Apple in un potente sistema di scrittura (word Processor). Collegando ad Apple una stampante a margherita (per un carattere perfetto) oppure una stampante veloce (Rosy 26) potrete produrre con enorme efficienza testi ripetitivi, contratti, offerte, circolari, etc. Richiede 48K, 1 disco, 1 Stampante

EASYWRITER L. 95.000 + IVA
Trattasi di un "word processor" molto sofisticato, con particolari istruzioni per l'incolonnamento, modifica, cancellazione, introduzione di nuove righe di testo, etc. Richiede 48K di memoria, 1 disco, 1 stampante.

APPLEPOST L. 79.000 + IVA
Se doveste frequentemente spedire materiale pubblicitario o di documentazione a centinaia di indirizzi, Applepost risolverà brillantemente il problema di arichiviare, aggiornare, stampare Etichette.
Richede 48K, 1 disco, 1 stampante.

PROGRAMMI DI UTILITÀ

MICROMEMO L. 79.000 + IVA
Il Vostro Apple si trasforma con questo programma in una potente agenda computerizzata, ricorda appuntamenti, scadenze, pagamenti, note personali. Per un intero anno i Vs. impegni personali o di lavoro vengonomemorizzati da questo programma, permettendoVi una rapida consultazione. Richiede 48K, 1 disco.

RICOPIA DISCHETTI CON SINGOLO DRIVER L. 150.000 + IVA
Se volete "mettere al sicuro" i Vs. programmi e quindi il Vs. lavoro, pur possedendo un Apple con 1 solo disk-driver, potrete ricopiare i Vs. dischetti con questo utilissimo programma, che non è indispensabile per il Vs. lavoro.
Richiede 48K di memoria, 1 disco.

RUBRICA TELEFONICA INTELLIGENTE L. 30.000 + IVA
La ricerca di un nome o di un numero ricordandosi soltanto parzialmente uno di essi vene effettuata in pochi istanti da questo interessante programma. È un potente aiuto per la Vs. memoria, dato che per la ricerca basta una sola sillaba, al resto ci pensa il Vs. Apple.
Contiene 150 numeri (può essere estesa sino a 1000). Nella versione minima richede 16K, × disco.

SOME COMMON BASIC PROGRAMS L. 50.000 + IVA
Trattasi di un set di programmi per calcoli finanziari, satistica, diagnostica, miscellaneus (varie). Richiede 48K, 1 disco

CONTRIBUTOR'S PROFRAMS VOL. 1/2 L.30.000 + IVA
Programmi applicativi per i campi più svariati. Sono di grande aiuto al programmatore in Basic Applesoft. Richiede 48K, 1 disco.

CONTRIBUTOR'S PROGRAMS VOL. 3/4/5 L. 45.000 + IVA
Logico seguito al n 1 ed al numero 2. richiede 48K, 1 disco

APPLESCACCHI MICROCHESS L. 30.000 + IVA
È molto difficile battere il Computer, lui non ha ne sviste ne indecisioni: comunque è possibile, prevedendo almeno 80 mosse.

FUNCTION PLOT HI RESOLUTION L. 50.000 + IVA
Come realizzare molte funzioni grafiche con Apple. 48K, 1 disco

CAPABILITIES DEMO DISKETTE L. 50.000 + IVA
Dimostrativo delle capacità grafiche di Apple. 48K, 1 disco.

DIMOSTRATIVO DI INGEGNERIA CIVILE L. 50.000 + IVA
In PASCAL UCDS - Disponibile da Giugno 80.

PROGRAMMA DI DUPLICAZIONE DISCHETTI L. 35.000 + IVA
In Italiano. richiede 48K, 2 Dischi.

I ns. pakages sono duplicabili con i 2 programmi di duplicazione in elenco

Comunicato n. 4 ai Rivenditori di Macchine per Ufficio.

Unicomp iBEX 7000: o li vendete voi, o li vendono gli altri.

Unicomp iBEX 7000: i sistemi gestionali forti della tecnologia giapponese.

La serie iBEX 7000 è una famiglia di sistemi gestionali di produzione giapponese: il massimo per quanto riguarda la qualità tecnologica.

E in più hanno tutto per garantire un sicuro, grande successo commerciale:

- alte prestazioni;
- un prezzo molto interessante;
- grande robustezza e affidabilità;
- un design piacevolmente pulito e funzionale.

Con la Unicomp cominciate a vendere subito. E tutto quel che vi serve è un limitato investimento.

La Unicomp vi permette di limitare al massimo gli investimenti necessari per cominciare a vendere sistemi Cromemco e altri personal computers.

Perché, oltre alle macchine, noi vi diamo una gamma di servizi che non ha l'uguale in questo settore.

- Tutto l'addestramento che serve ai vostri tecnici, per metterli in grado di identificare eventuali guasti a livello di scheda.
- Un laboratorio riccamente attrezzato e con personale altamente qualificato per riparare, se lo desiderate, le schede difettose o eventuali unità periferiche.
- Parti di ricambio sempre disponibili e una documentazione sempre aggiornata, per permettervi di offrire ai vostri clienti un servizio di assistenza professionale ed efficiente.
- Addestramento dei vostri programmatori sul software di base.
- Raccolta di programmi standard: contabilità generale, contabilità IVA, clienti e fornitori, gestione di magazzino, fatturazione, etc. La disponibilità di questi programmi, cosa molto importante per ridurre i vostri investimenti nello sviluppo di procedure applicative, vi mette subito in grado di eseguire dimostrazioni e di accettare ordini dai vostri clienti. E su questi programmi, concepiti per essere uno strumento nelle vostre mani, potete intervenire facilmente per modificarli e renderli "su misura" delle specifiche esigenze dei vostri clienti.

Piccole aziende, professionisti, commercianti e molti altri operatori possono essere certi che il loro investimento su un iBEX 7000 vale nel tempo e si traduce giorno per giorno in prezioso supporto per il loro lavoro.

La famiglia degli iBEX 7000 comprende diversi modelli. La versione 7100 contiene due mini-floppy da 280 KByte ciascuno, e memoria da 64K. La versione 7200 presenta due floppy da 8" compatibili IBM, con capacità di 1 MB ciascuno (fino a quattro collegabili), e memorie di 64K.

I sistemi sono basati sul popolare microprocessore Z80 e sono CP/M compatibili: possono quindi usare tutte le procedure gestionali della Unicomp. È inoltre disponibile un efficiente sistema di word-processing.

UNICOMP

i piccoli sistemi di grande avvenire.

20092 cinisello balsamo (milano)
palazzo testi / via cantù, 20
tel.(02) 6121041 (5 linee)

COMPUTERIA® Il Centro del Personal Computer — è marchio registrato della Unicomp S.r.l.

guida mercato computer

Nota: le schede tecniche complete di tutti gli apparecchi saranno pubblicate per intero nell'ANNUARIO di m&p COMPUTER.

AGGIORNAMENTO

PERSONAL COMPUTER

Prodotto	Prezzo	Scheda tecnica apparsa sul numero	Riferimento servizio lettori
ALTOS (USA) *Ediconsult S.r.l. - Via Rosmini, 3 - Monza*			
ACS 8000-1 32 K RAM; 2 floppy disk singola densità, singola faccia	L. 5.000.000 + IVA	3	100
ALTOS (USA) *Segi - distrib. NORDITALIA - Via Timavo, 12 - Milano* *Microcomp - distrib. CENTROSUD - Via M. Gelsomini, 28 Roma*			
ACS - 8000-1 64 K RAM, 2 floppy disk singola densità, singola faccia (prezzo riferito al cambio del dollaro L. 900)	L. 5.870.000 IVA	3	101
APPLE COMPUTER INC. (USA) *Iret - Via Emilia S. Stefano, 32 - Reggio Emilia*			
Apple II 16 K RAM	L. 1.904.400 IVA comp.	3	102
Apple II 32 K RAM	L. 2.005.600 IVA comp.	3	
Scheda colore PAL	L. 275.300 IVA comp.		
Language System/Pascal	L. 739.000 IVA comp.		
Interfaccia parallela (Centronics)	L. 318.000 IVA comp.		
Interfaccia seriale	L. 275.000 IVA comp.		
Floppy disk con controller	L. 914.250 IVA comp.		
Secondo Driver per floppy disk	L. 816.500 IVA comp.		
Tavoletta grafica	L. 1.182.000 IVA comp.	5	103
Monitor 12' fosfori verdi	L. 472.000 IVA comp.		104
Monitor 9' fosfori verdi	L. 330.000 IVA comp.		105
ASEL (ITALIA) *ASEL - Via Cortina d'Ampezzo, 17/A - Milano*			
Scheda base Amico 2000	L. 305.000 + IVA	5	106
Contenitore completo di interfacce e alimentatore di potenza	L. 350.000 + IVA	5	
Tastiera ASCII	L. 144.000 + IVA	5	
Espansione RAM/ROM con BASIC standard	L. 299.000 + IVA		
Interfaccia video	L. 249.000 + IVA		
Scheda RAM 16 K	L. 319.000 + IVA		
Scheda RAM 32 K	L. 419.000 + IVA		
Scheda I/O seriale RS-232 e parallela	L. 154.000 + IVA		
Scheda interfaccia floppy disk	L. 299.000 + IVA		
ATARI (USA) *Adveico - Via Emilia Ovest, 123 - S. Pancrazio (Pr)*			
Atari 400 Computer	L. Annunciato	-	107
Atari 800 Computer	L. Annunciato		108
COMMODORE (USA) *Harden S.p.A. - Divisione Elettronica-Sospiro (Cr)*			
PET 2001 8 K RAM	L. 934.500 + IVA	3	109
PET 3032 32 K RAM	L. 1.765.000 + IVA	3	110
PET 3032 + dual floppy + stampante 3022 con tractor feed	L. 6.300.000 + IVA	3	
Come il precedente ma con stampante Honeywell 132 colonne	L. 7.450.000 + IVA	3	
COMPUCOLOR CORPORATION (USA) *Compitant - Viale Michelangelo - Menfi (Ag)*			
Compucolor Executive 16 K, tastiera espansa	L. 3.895.000 + IVA	6	111
Compucolor Executive 32 K, tastiera espansa	L. 4.250.000 + IVA	6	
Compucolor II Mod. 4 16 K RAM	L. 2.790.000 + IVA	3	112
Compucolor II Mod. 5 32 K RAM	L. 3.240.000 + IVA	3	
Driver floppy aggiuntivo	L. 750.000 + IVA	3	

AGGIORNAMENTO

			Scheda tecnica apparsa sul numero	Riferimento servizio lettori
EXIDY COMPUTER SYSTEMS (USA)				
Unicomp divisione computeria - Palazzo Testi				
Via Cantù, 20 - Cinisello Balsamo (Mi)				
Sorcerer 8 K RAM	L.	1.000.000 + IVA	3	113
Sorcerer 16 K RAM	L.	1.250.000 + IVA	3	
Sorcerer 32 K RAM	L.	1.500.000 + IVA	3	
Sorcerer 48 K RAM	L.	1.750.000 + IVA	3	
Doppio driver floppy disk con controller	L.	3.250.000 + IVA		
Monitor televisivo	L.	680.000 + IVA		
Espansione Bus S-100	L.	595.000 + IVA		
GENERAL PROCESSOR (ITALIA)				
General Processor - Via Pian dei Carpini, 1 - Firenze				
T/08-21 con doppio floppy disk singola densità, 16 K RAM	L.	4.088.000 + IVA	3	114
T/08-22 con doppio floppy disk doppia densità, 32 K RAM	L.	4.247.000 + IVA	3	
T/10-2 con doppio floppy disk IBM/2 side	L.	6.159.000 + IVA		
T/20 con disco fisso 10 Mbytes + Mbyte floppy 8"48 K RAM interfaccia stampante	L.	12.800.000 + IVA		
Stampanti a partire da	L.	1.098.000 + IVA		
Espansione RAM di 16 K	L.	259.000 + IVA		
Interfaccia stampante,	L.	259.000 + IVA		
08-222 con doppio floppy disk doppia faccia doppia densità (560 K)	L.	4.950.000 + IVA		
HEATH (USA)				
Larir - Viale Premuda, 38/A - Milano				
WH8 Computer	L.	741.000 + IVA	3	115
H8-2 Interfaccia parallela	L.	349.000 + IVA		
WH8-16 Memoria 16 K	L.	642.000 + IVA		
WH8-5 Interfaccia seriale e per cassette	L.	223.000 + IVA		
WH17 Unità floppy disk con controller	L.	1.198.000 + IVA		
WH17 1 Second driver floppy disk	L.	756.000 + IVA		
WH19 Video terminale	L.	1.850.000 + IVA		
WH14 Stampante	L.	1.664.000 + IVA		
HC8-14 Extended BASIC su cassetta	L.	55.700 + IVA		
H8-17 Sistema operativo per mini floppy	L.	349.000 + IVA		
WH-11A Computer 16 bit	L.	3.708.000 + IVA	3	116
H 11-6 Chip aritmetico	L.	441.000 + IVA		
WH19 Terminale video	L.	1.850.000 + IVA		
WH14 Stampante	L.	1.664.000 + IVA		
H36 DEC writer LA 36	L.	2.780.000 + IVA		
WH27 Doppio driver floppy disk	L.	4.824.000 + IVA		
HT 11 Sistema operativo con BASIC	L.	650.000 + IVA		
HT 11-1 FORTRAN	L.	465.000 + IVA		
WH89 Computer	L.	4.050.000 + IVA	3	117
H8-17 Software sistema operativo	L.	349.000 + IVA		
H8-21 Microsoft BASIC	L.	349.000 + IVA		
HEWLETT PACKARD (USA)				
Hewlett Packard Italiana - Via G. di Vittorio, 9 - Cernusco sul Naviglio (Mi)				
HP-85 Computer	L.	4.160.000 + IVA	3	118
Plotter 7225 A	L.	3.869.000 + IVA		119
16 K RAM	L.	505.000 + IVA		
Interfaccia HPIB	L.	505.000 + IVA		
Cassetta ROM	L.	58.000 + IVA		
ROM I/O	L.	365.800 + IVA		
ROM Print Plot	L.	179.800 + IVA		
ROM Matrix	L.	179.800 + IVA		
Stampante	L.	4.983.000 + IVA		120
INTERTEC DATA SYSTEM (USA)				
Cattaneo System S.r.l. - Via Caffaro, 2a - Genova				
Superbrain 32 K, CP/M, BASIC	L.	4.900.000 + IVA	4	121
Superbrain 64 K, CP/M, BASIC	L.	5.400.000 + IVA	4	
Compilatore COBOL + documentazione	L.	1.320.000 + IVA		
Compilatore FORTRAN + Documentazione	L.	900.000 + IVA		

	Prezzo	Scheda tecnica apparsa sul numero	Riferimento servizio lettori
Superbrain QD 64 K RAM 700 Kbyte, CP/M. BASIC	L. 6.150.000 + IVA		122

INTERTEC DATA SYSTEM (USA)
Seimar Computer - Galleria del Corso, 4 - Milano

	Prezzo	Scheda tecnica apparsa sul numero	Riferimento servizio lettori
Superbrain 32 K RAM, CP/M, BASIC	L. 4.700.000 + IVA	4	123
Superbrain 64 K RAM, CP/M, BASIC	L. 5.200.000 + IVA	4	
Superbrain 64 K, CP/M, BASIC compiler	L. 5.400.000 + IVA		
Superbrain 64 K, CP/M, COBOL compiler	L. 6.000.000 + IVA		

INTERTEC DATA SYSTEM (USA)
SMC - Via Settimio Mobilio, 23 - Salerno

	Prezzo	Scheda tecnica apparsa sul numero	Riferimento servizio lettori
Superbrain 32 K RAM	L. 3.980.000 + IVA (OEM)	4	124
Espansione 32 K RAM	L. 600.000 + IVA (OEM)	4	
Linguaggio BASIC	L. 700.000 + IVA (OEM)		
Adattatore Bus S-100	L. 1.300.000 + IVA (OEM)		
Linguaggio FORTRAN	L. 1.000.000 + IVA (OEM)		

LORENZON ELETTRONICA (ITALIA)
Lorenzon Elettronica S.n.c. - Via Venezia, 115 - Oriago di Mira (Ve)

	Prezzo	Scheda tecnica apparsa sul numero	Riferimento servizio lettori
CTL nuova versione con video e fosfori verdi	L. 1.798.000 + IVA	4	125
Interfaccia registratori a cassette	L. 89.000 + IVA		
Scheda I/O parallelo	L. 64.000 + IVA		
Scheda I/O seriale	L. 80.000 + IVA		
Espansione 8 K RAM	L. 198.000 + IVA		
Interfaccia stampante	L. 100.000 + IVA		
Doppio floppy disk 160 K	L. 1.798.000 + IVA		

MOTOROLA (USA)
Motorola S.p.A. Divisione Semiconduttori - Via Ciro Menotti, 11 - Milano

	Prezzo	Scheda tecnica apparsa sul numero	Riferimento servizio lettori
EXORset 30	L. 5.830.000 + IVA	6	126

NORTH STAR COMPUTERS (USA)
Zelco - Via V. Monti, 21 - Milano

	Prezzo	Scheda tecnica apparsa sul numero	Riferimento servizio lettori
Horizon Computer con 2 driver doppia densità 32 K RAM, I/O parallela e seriale, alimentatore.	L. 3.310.000 + IVA	3	127
Come sopra ma con 48 K RAM	L. 3.845.000 + IVA		
Come sopra ma con 64 K RAM	L. 4.300.000 + IVA		
Video IQ 20	L. 1.380.000 + IVA		
Stampante 80 col.	L. 1.090.000 + IVA		
Sistema: Horizon 64 K, disco rigido 18 Mbyte	L. 10.580.000 + IVA		

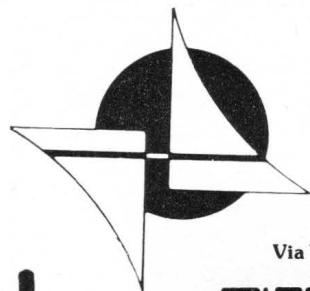
OHIO SCIENTIFIC (USA)
Ediconsult - Via Rosmini, 3 - Monza

	Prezzo	Scheda tecnica apparsa sul numero	Riferimento servizio lettori
Challenger 1 P Computer con BASIC da 8 K e 4 K RAM	L. 651.000 + IVA (OEM)	3	128

PLAE (ITALIA)
Plae - Via Curtatone, 16 - S. Giuliano Milanese (Mi)

	Prezzo	Scheda tecnica apparsa sul numero	Riferimento servizio lettori
Alpha 1 Computer	L. 1.150.000 + IVA	3	129

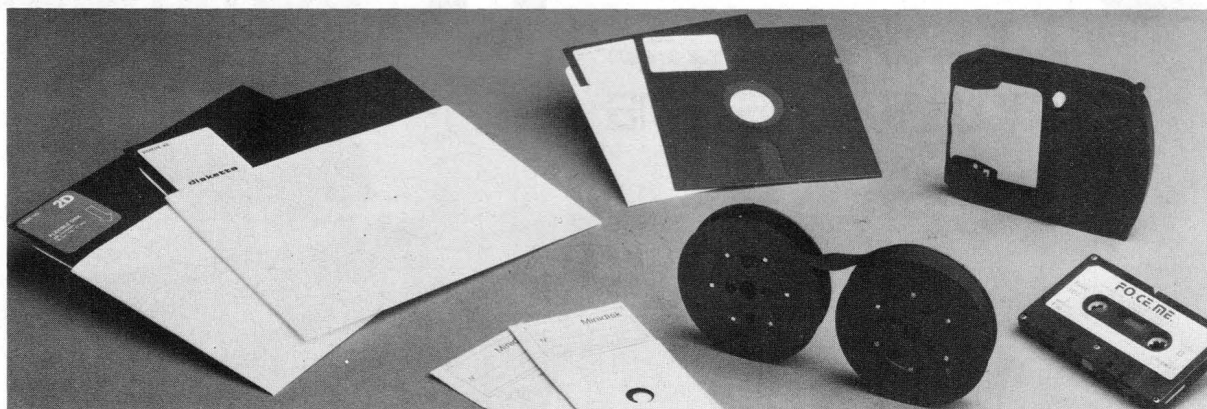
RADIO SHACK (USA)
Radio Shack Italia - Corso Europa, 12 - Milano

	Prezzo	Scheda tecnica apparsa sul numero	Riferimento servizio lettori
TRS 80 Modello I, livello 1 4 K RAM	L. 995.000 + IVA	3	130
TRS 80 Modello I, livello 2 4 K RAM	L. 1.166.000 + IVA	3	
TRS 80 Modello I, livello 2 16 K RAM	L. 1.675.000 + IVA	3	
Interfaccia O K	L. 507.000 + IVA	3	
Interfaccia 16 K	L. 896.000 + IVA	3	
Interfaccia 32 K	L. 1.340.000 + IVA	3	
Primo driver per floppy	L. 852.000 + IVA		
Driver successivi	L. 829.000 + IVA		
Stampante 2611/56	L. 2.048.000 + IVA		
TRS 80 Modello II, 32 K RAM, 4 driver floppy disk 8" 2 Mbyte in linea, stampante 80 colonne, cavi di collegamento	L. 9.270.000 + IVA	4	131

AGGIORNAMENTO

	Prezzo	Scheda tecnica apparsa sul numero	Riferimento servizio lettori
Come sopra ma con stampante 132 colonne	L. 10.280.000 + IVA		
Come sopra ma con 64 K RAM e stampante 80 colonne	L. 9.980.000 + IVA		
Come sopra ma con stampante 132 colonne	L. 10.990.000 + IVA		
TRS-80 II 32 K, 1 floppy disk, cavi di collegamento	L. 5.290.000 + IVA		
1145 interfaccia RS-232	L. 179.000 + IVA		
1180 Sintetizzatore vocale	L. 640.000 + IVA		132
1181 Controllo vocale	L. 339.000 + IVA		133
Interfaccia telefonica 2 vie (modem)	L. 423.000 + IVA		134
SD SYSTEM (USA) *Computer Company - Via S. Giacomo, 32 - Napoli*			
SD 200 Computer 64 K RAM, 2 Mbytes di memoria di massa	L. 8.800.000 + IVA	4	135
Stampante 120 caratteri per secondo	L. 1.900.000 + IVA		
SHARP CORPORATION (GIAPPONE) *Melchioni Computer Time - Via P. Coletta, 37 - Milano*			
MZ-80K Computer 20 K	L. 1.540.000 + IVA	3	136
MZ-80K Computer 48 K	L. 1.960.000 + IVA	3	
MZ-80K Computer 48 K + stampante (Sharp) 80 colonne + interfaccia I/O	L. 3.725.000 + IVA		
SOUTH WEST TECHNICAL PRODUCT CORPORATION (USA) *Homic - Piazza de Angelis, 1 - Milano*			
SWTPC 6809 Computer 56 K RAM, videoterminale stampante 132 colonne, doppio driver floppy disk da 2.5 Mbytes	L. 12.500.000 + IVA	3	137
TEXAS INSTRUMENTS (USA) *Texas Instruments Semiconduttori Italia - Cittaducale (Ri)*			
TI 99/4 Computer	L. Annunciato	3	138

	Prezzo	Scheda tecnica apparsa sul numero	Riferimento servizio lettori
VECTOR GRAPHIC (USA) *C.d.S. - Scali d'Azeglio, 52 - Livorno*			
Computer V.I.P., Video console 12", 64 K dinamici, 1 floppy 5" CP/M-2, stampante Centronics	L. 7.300.000 + IVA		139
WESTERN DIGITAL (USA) *Comprel - Viale Romagna, 1 - Cinisello Balsamo (Mi)*			
WD/90 Pascal Microengine	L. 3.820.000 + IVA	3	140
ZENITH DATA SYSTEMS (USA) *Adveico - Via Emilia Ovest, 129 - S. Pancrazio (Pr.)*			
Z-89 Computer 48K, CP/M 2.2, Microsoft BASIC	L. 4.128.000 IVA comp.		141
ZILOG (USA) *Zelco - Via V. Monti, 21 - Milano*			
MCZ 1-05 Computer	L. 7.200.000 + IVA		142
MCZ Computer + Video IQ 20 + stampante Centronics 730	L. 9.500.000 + IVA	6	
MCZ 1-20 Computer	L. 8.500.000 + IVA		143

STAMPANTI

	Prezzo	Scheda tecnica apparsa sul numero	Riferimento servizio lettori
BASE 2' INC. *Compitant - Viale Michelangelo - Menfi (Ag)*			
800 Impact printer	L. 950.000 + IVA	5	144
Stepper motor per plotting	L. 75.000 + IVA		145
CENTRONICS *Centronics Data Computer - Via S. Valeria, 5 - Milano*			
Modello 700	L. 1.800.000 + IVA	5	146
Modello 701	L. 2.000.000 + IVA	5	147
Modello 702	L. 2.500.000 + IVA	5	148
Modello 737	L. 1.200.000 + IVA	5	149

			Scheda tecnica apparsa sul numero	Riferimento servizio lettori
Modello 779	L.	1.350.000 + IVA	5	150
Modello 730-2	L.	1.000.000 + IVA	5	151
Modello 730-4	L.	1.100.000 + IVA	5	152
COMMODORE (USA)				
Harden S.p.A. - Divisione Elettronica - Sospiro (Cr)				
Model 3022	L.	1.040.000 + IVA	5	153
EPSON				
SEGI - Via Timavo, 12 - Milano				
Modello TX-80	L.	850.000 + IVA	5	154
FACIT				
Facit Data Products S.p.A. - Via Toffetti, 2 - Milano				
Modello 4520	L.	1.100.000 + IVA	5	155
HONEYWELL				
Microlemdata S.r.l. - Via Pellizzani, 29 - Milano				
S 10 (10 unità)	L.	850.000 + IVA	5	156
HONEYWELL				
Harden S.p.A. - Divisione Elettronica - Sospiro (Cr)				
Lina 20 (interfaccia per il PET)	L.	2.050.000 + IVA	5	157
OKI				
Technitron - Via Mangili, 20 - Roma				
Microline 80 interfaccia parallela	L.	850.000 + IVA	5	158
Tractor feed	L.	90.000 + IVA		
Interfaccia seriale RS-232	L.	150.000 + IVA		
DP-125	L.	2.700.000 + IVA	5	159
DP-160	L.	3.500.000 + IVA	5	160
DP-250	L.	3.700.000 + IVA	5	161
DP-300	L.	4.200.000 + IVA	5	162
Interfaccia seriale	L.	280.000 + IVA	5	
Opzione grafica	L.	350.000 + IVA	5	

			Scheda tecnica apparsa sul numero	Riferimento servizio lettori
RADIO SHACK (USA)				
Radio Shack Italia - Corso Europa, 12 - Milano				
Quick II 11/55	L.	329.000 + IVA	5	163
Line printer II 11/54	L.	998.000 + IVA	5	164
Quick printer 11/53	L.	798.000 + IVA	5	165
Line printer III	L.	2.048.000 + IVA	5	166
Stampante Trattore 80 o 132 car.	L.	1.690.000 + IVA		167
TRENDCOM				
Telcom - Via Matteo Civitali, 75 - Milano				
Modello 200	L.	841.500 + IVA	5	168
Modello 100	L.	527.000 + IVA	5	169

SCHEDE MICROCOMPUTER

ASEL (ITALIA)				
Asel S.r.l. - Via Cortina d'Ampezzo, 17 - Milano				
Amico 2000 A scheda base (montato)	L.	305.000 + IVA	3	170
E & L INSTRUMENT (USA)				
Microlem S.a.s. - Via Monteverdi, 5 - Milano				
MMD-1 scheda computer (montato)	L.	390.000 + IVA	3	171
MMD-1 scheda computer (Kit)	L.	315.000 + IVA	3	
EMMECI (ITALIA)				
L'Emmeci - Via Porpora, 132 - Milano				
MMS-8 Livello 1, scheda base, consolle esadecimale, alimentatore	L.	350.000 + IVA	3	172
MOS TECHNOLOY (USA)				
Skylab S.r.l. - Via M. Gioia, 66 - Milano				
KIM-1 scheda base (prezzo riferito al cambio del dollaro L. 840)	L.	250.000 + IVA	3	173

AGGIORNAMENTO

			Scheda tecnica apparsa sul numero	Riferimento servizio lettori
NASCOM MICROCOMPUTER (GRAN BRETAGNA) *Homic - Piazza de Angelis, 1 - Milano*				
Nascom-1 (Kit)	L.	480.000 + IVA	3	174
Nascom-1 (montato)	L.	560.000 + IVA	3	
Scheda Buffer (Kit)	L.	105.000 + IVA	3	
Espansione di memoria 16 K RAM (Kit)	L.	390.000 + IVA	3	
Nascom-2 8 K RAM (Kit)	L.	1.100.000 + IVA		175
Nascom-2 16 K RAM dinamici (Kit)	L.	1.200.000 + IVA		
Nascom-2 8 K (montato)	L.	1.320.000 + IVA		
Nascom-2 16 K dinamici (montato)	L.	1.460.000 + IVA		
ROCKWELL INTERNATIONAL (USA) *Ing. De Mico - Via Manzoni, 31 - Milano*				
AIM-65 1 K RAM	L.	535.000 + IVA	3	176
AIM-65 4 K RAM	L.	595.000 + IVA	3	
Assembler 4 K	L.	119.000 + IVA	3	
BASIC 8 K	L.	140.000 + IVA	3	
Programmatore di EPROM	L.	95.000 + IVA	4	
Alimentatore	L.	80.000 + IVA		
Espansione 16 K RAM	L.	545.000 + IVA		
Interfaccia per TV (64X16)	L.	345.000 + IVA		
Minifloppy disk controller	L.	345.000 + IVA		
SGS (ITALIA) *SGS ATES Componenti Elettronici S.p.A. - Via C. Olivetti, 2 - Agrate Brianza (Mi)*				
NBZ 80 Scheda microcomputer	L.	471.000 + IVA	3	177
NPZ 80 Alimentatore	L.	165.000 + IVA	3	
UPZ-80BS Scheda esperimenti	L.	306.000 + IVA	3	
SYNERTEC SYSTEM CORPORATION (USA) *Comprel - Viale Romagna, 1 - Cinisello Balsamo (Mi)*				
SYM-1 Scheda microcomputer	L.	320.000 + IVA	3	178
TEXAS INSTRUMENTS (USA) *Texas Instruments Semiconduttori Italia - Cittaducale (Ri)*				
TM 990/189 M Scheda base microcomputer	L.	415.000 + IVA	3	179

CALCOLATRICI PROGRAMMABILI

CASIO *Ditron - Via Certosa, 138 - Milano*				
FX 502 P	L.	199.000 + IVA	3	180
HEWLETT PACKARD *H. P. Italiana - Via G. Di Vittorio, 9 - Cernusco sul Naviglio (Mi)*				
HP 33E	L.	108.000 + IVA	3	181
HP 33C	L.	132.000 + IVA	3	182
HP 34C	L.	184.000 + IVA	3	183
HP 38E	L.	147.000 + IVA	3	184
HP 38C	L.	184.000 + IVA	3	185
HP 97A	L.	870.000 + IVA	3	186
HP 97S	L.	1.552.000 + IVA	3	187
HP 67	L.	422.000 + IVA	3	188
HP 41C	L.	374.000 + IVA	3	189
Lettore di schede	L.	273.000 + IVA		
Espansione di memoria	L.	55.800 + IVA		
Moduli programmati	L.	55.800 + IVA		
Stampante	L.	488.000 + IVA		
Lettore ottico	L.	158.000 + IVA		
SHARP *Melchioni - Via P. Coletta, 37 - Milano*				
EL 5100	L.	134.900 + IVA	3	190
PC 1211	L.	259.500 + IVA	7	191
Interfaccia registratore	L.	31.500 + IVA		
TEXAS INSTRUMENTS *Texas Instruments Semiconduttori Italia - Cittaducale (Ri)*				
TI 57	L.	49.000 + IVA	3	192
TI 58	L.	129.000 + IVA	3	193
TI 58 C	L.	139.000 + IVA	3	194
TI 59	L.	248.000 + IVA	3	195
PC 100 C	L.	289.000 + IVA	3	196

Tandy Radio Shack ITALIA

- il più venduto computer nel mondo — **TRS - 80**
- il più economico — **TRS - 80**
- il più assistito — **TRS - 80**
- sistemi chiavi in mano — GESTIONALI - AMMINISTRATIVI SCIENTIFICI

L'UNICO CONCESSIONARIO
L'UNICO CON GARANZIA

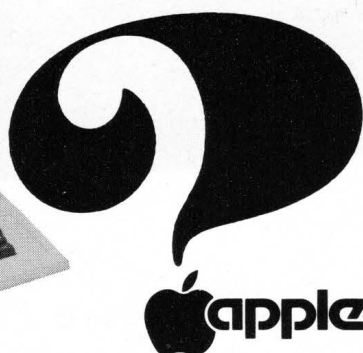
FIRENZE
VIA SESTESE, 22/24
TEL. (055) 45.43.19

CONSULENZA QUALIFICATA
VENDITA
ASSISTENZA TECNICA

- Periferiche di ogni tipo
- Interfacce standard e su misura
- Stampanti per ogni esigenza (CENTRONICS, HONEYWELL, ecc.)

SERCOM - **BOLOGNA - ☎ (051) 478.539**
Via Berengario da Carpi, 9/B

il negozio in fatto di . . . mele

a Bologna zona Palasport
via Ercolani 10/D tel. (051) 23 98 57

80 mq. a vostra disposizione

apple-core club — hardware
software — consulenza edp

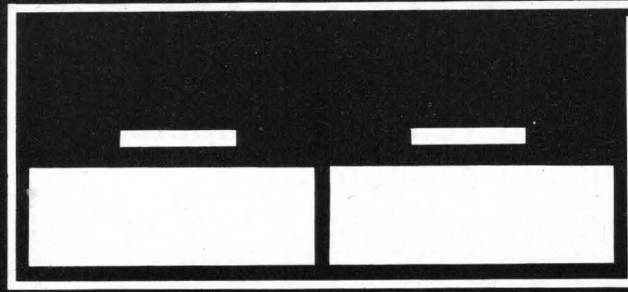
apple computer

Distribuzione per l'Italia
IRET®
informatica

Titolare Dott. Massimo Fantuzzi rivenditore autorizzato

GAMMA COMPUTER s.r.l.

Un computer per
tutte le esigenze:

PET
APPLE
ALTOS
HP
STAMPANTI CENTRONICS - HONEYWELL

Distributore «SEGI» per Sicilia e Calabria

VISITATECI

Troverete la risposta
ad ogni vostro problema

Sede Catania - Via Gabriele D'Annunzio, 125 - Tel. 095/386954
Sede di Siracusa - Via G. Di Natale, 14 - Tel. 0931/68820

COMPUTERCOMPROVENDO

I piccoli annunci dei Lettori (massimo 50 parole) sono pubblicati gratuitamente. Le prime due parole dell'annuncio verranno pubblicate in neretto. Saranno cestinate le inserzioni chiaramente a carattere commerciale o speculativo e quelle anonime (tipo fermo posta), per non favorire attività illecite. Preghiamo gli interessati di inviare solo annunci che abbiano come oggetto materiali attinenti l'argomento trattato dalla rivista.
È possibile la pubblicazione di annunci a pagamento, interamente in neretto (corpo 8), al prezzo di L. 4.000 (quattromila) per riga tipografica (circa 45 caratteri riga).

TI-58 come nuova, con modulo di Base e Statistico, **vendo** L. 130.000. Roberto Gradnik - Via Lanzone 2 - MILANO - Tel. 807338 ore serali.

RAM HP 41C vendo due a L. 55.000 ognuna, trattabili, anche separatamente. Daniele Stella - Via Montenero, 47 - 48100 Ravenna - Tel. 0544/32529.

TI 58 C in perfetto stato, ancora in garanzia, completa di modulo di base, alimentatore-ricaricatore, mascherine per programmi, istruzioni, **vendo** a L. 100.000. Luciano Cuneo - Via Emilio Lepido, 46 - 00175 Roma - Tel. 06/7491542

Software HP 41 C per ingegneria civile: analisi strutturale - cemento armato agli stati limite - C.A.P. - strutture in acciaio - geotecnica e fondazioni - ingegneria idraulica. Il massimo fattibile su HP 41C è già disponibile. Studio Ing. Paolo Arnaud - Via Nino Costa - 14100 ASTI - Tel. 0141/32484.

Vendo personal computer 8K RAM utente + 8K ROM BASIC esteso e veloce; tastiera ASCII standard, interfaccia video, cassette, stampante seriale. Espandibile a 32K RAM. Tre mesi vita, usato tre volte, passo a sistema superiore. Richiedo 55% prezzo commerciale cioè L. 880.000 trattabili. Dott. Salvatore Bellini - Via Tasso, 96 - Pavia.

TI-58, HP 67 vendo.
Dotazioni standard L. 100.000; L. 370.000. Scrivere a: Dominici Roberto - Via Quintilio Varo, 133 - 00174 ROMA

TM 990/189 M TEXAS I solo zona di Torino **vendo**, mai usato completo di **2K RAM** e manuali d'uso. Telefonare ore serali 011/939238.

HP 34 C HEWLETT PACKARD cedo al miglior offerente, acquistata luglio '80, mai usata, completa di alimentatore e manuali, in garanzia. Beppe Fasolis - Piazza San Brunone, 1 - 14100 Asti - Tel. 0141/55892.

TI 59 + stampante, custodie, rotoli carta, schede, **vendo** a L. 350.000 + IVA. Audiovideal srl, Umberto Santucci - Via Spallanzani, 32 - 00161 Roma - Tel. 06/858325.

HP 65 vendo, completa di accessori, ricca di programmi di matematica, macchine design, finanza, scambio calore e massa, ingegneria chimica. Ottima per studenti, a prezzo imbattibile (incluse circa 200 schede magnetiche con relativa letteratura descrittiva completa di esempi). M. Maiorano - Via Madonna 7 Dolori, 12 - 66054 Vasto (CH).

Programmi per TI-59/TI-58 (C) con o senza stampante PC-100. Informazioni inedite rispetto al manuale di istruzione! Richiedere elenco programmi e fogli informativi, allegando 1000 lire (anche in francobolli) per spese materiale e postali. Servizio di programmazione su richiesta.
Paolo Capobussi - Via Pisani Dossi, 27 - 20134 Milano

Vendo i seguenti **programmi per TI-59:**
1) rubrica telefonica (non serve stampante!) L. 15.000.
2) Gioco delle «Inversioni» + gioco del «Filetto» L. 10.000.
Luca De Matteis - V.le S. Lavagnini, 26 - 50129 Firenze.

Vendo TRS-80 livello 2 4K praticamente nuovo, completo monitor, tastiera, registratore con software home-utility L. 1.000.000.
Ariotti Donatella - Tel. 0571/61554 (sabato oppure ore pasti)

Vendesi microelaboratore **«Amico 2000A»** collaudato funzionante con contenitore - comprendente interfaccia cassette - tastiera video - interfaccia video - EPROM di gestione - scheda mini BASIC - Valore L. 1.250.000 prezzo richiesto L. 800.000. Telefonare ore ufficio 418.641 Sig.ra Mimma.

HP 33E programmabile **vendo,** 9 mesi di vita, ancora in garanzia, completa di manuali di uso e applicazioni. All'acquirente regalo inoltre biblioteca programmi riguardanti la matematica (Newton, integrati, numeri complessi, matrici, vettori, etc.). Il tutto in imballaggio originale a L. 110.000. Scrivere o telefonare a: Giampiero De Giovanni - Via Ferrara, 30/33 - 16127 Genova - Tel. 010/261475.

SYM 1 - Praticamente nuovo, 8K BASIC, 4K RAM, 58 linee I/O, espansione 24K RAM, terminale video (verde) con grafica, reverse e controllo cursore, in elegante contenitore. Ho anche 40 memorie 2114 e molti giochi grafici su cassetta. **EPROM PROGRAMMER** con istruzioni. **Vendo** anche a pezzi separati.
Marchetto Alessandro - Via dei Cignoli, 8 - 20151 Milano - Tel. 306403.

Vendo Apple - 48K + programmi vari L. 1.500.000, 2 unità disco con controller + programma Data Base L. 1.300.000, stampante Centronics 730 + interfaccia + programma «word processing» L. 1.000.000. Tutto in perfetto stato (1 mese di vita). Telefonare 051/801292 e chiedere di Silvano.

Vendo TI 59 4 mesi di vita come nuova L. 280.000 trattabili. **Stampante P C100C** 2 mesi di vita come nuova L. 300.000 trattabili. **TI 57 e TI 51 II** 2 anni di vita usate molto poco - ottimo stato - L. 38.000 cadauna. - Tel. 011/6069807.

TEXAS SR-52 vendo, completa di tutti gli accessori, 22 schede della biblioteca base più 21 schede vergini e vari programmi di analisi matematica da me svolti, L. 150.000. **Vendo** anche **raccolta di programmi Texas** «Specialty pakettes matematics» L. 8.000. Stefano Gragnani - Via Italica, 36 - Lido di Camaiore (Lucca) - Tel. 0584/60226.

VENDO HP-41C completo di tre RAM e stampante HP alfanumerica con possibilità grafiche, in blocco L. 800.000. Carlo, ore serali 06/6160668.

Rockwell Aim 65, vendesi, fornito di memoria espansa, assembler, alimentatore, connettore per registrazione su cassetta + manuali in italiano. Come nuovo, ancora in garanzia, tre mesi di vita, perfettissimo in tutti i sensi, (ha ancora il rotolo di carta della casa). Costo effettivo circa 950.000 - (819.000 + IVA), ceduto a L. 850.000 trattabili. Bruno Gentile - Via Premuda n. 5 - Torino - Telefonare al 722264 (011).

Vendo manuale programmi 59 FUN originale **per TEXAS TI-59** con o senza stampante. Contiene giochi inediti; è corredato di esempi e listati, con le istruzioni per l'uso, molto chiari. Telefonare ore pasti Massimo Petrini 011/6964025. Strada Alberoni, 18/3 - 10133 Torino.

Teletype ASR 32 telescrivente a consolle. Motore sincrono 50 Hz, 45-50-75 baud, perforatore e lettore, tutto perfettamente funzionante completo dei tre manuali di servizio L. 650.000. Gianfranco Lucifora - Via Dodecanneso, 27/5 - 16146 Genova. Tel. 010/315340.

Cambio macchina fotografica professionale **Mamiya c 330** corredata di ob. 55 mm. e 80 mm. perfetta, **con AIM 65** (ev. conguaglio) oppure TI 59 + PC 100 C. Guccini Luciano - Via S. Francesco, 273 - 18011 Arma di Taggia (IM).

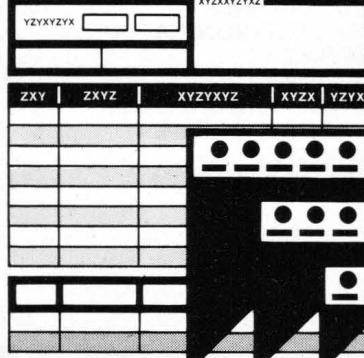
NOTIZIE APPLE

SILENTYPE

La stampante silenziosa, versatile e compatta che permette al Vostro Apple di produrre resoconti e grafici a 54.000 punti. Con la stessa semplicità Silentype vi permette di ottenere liste di programmi e stampe di testi con maiuscole e minuscole. Indispensabile nelle applicazioni scientifiche rappresenta un utile supporto nel campo professionale ed educativo, ove le caratteristiche grafiche ne mettono in risalto qualità ed adattabilità d'uso.
Silentype ed Interfaccia per Apple
L. 680.000 + IVA

NOTIZIE APPLE

VISICALC

Solo su Apple II il nuovo standard di qualità nel software per minisistemi.
Package completo per la gestione da video tramite cursore e scrolling di una enorme matrice di dati descrittivi (labels) e numerici. 254 righe e 63 colonne, Visicalc gestisce tutti i dati immessi nel sistema. Se un dato numerico viene variato da video dallo operatore, Visicalc rielabora tutti i dati associati al dato variato.
Sono ammessi campi alfanumerici fino a 31 caratteri. Insostituibile per previsioni aziendali, listini prezzi, bilanci. Visicalc non va programmato. È già pronto.

TRS 80 MODEL II 64K RAM + 1 Unità FLOPPY DISK esterna per un totale di 1 Megabyte in linea. NUOVISSIMI, perfettamente funzionanti, VENDO a Lire 6.900.000 trattabili. Scrivere o telefonare. Locatelli Andrea - Via Salnitrara, 3 - 43100 Parma - Tel. 0521/33559.

Vendo scheda AMICO 2000 completa di tutte le espansioni + alimentatore. Prezzo da concordarsi. Scrivere o telefonare ore 15-17 a: Giampiero Di Simone - Via Ignazio Sorrentino, 3 - 80059 Torre del Greco (NA) - Tel. 081/8821358.

Vendo libro «I Microprocessori in Pratica» (hardware, software, ricerca guasti) a lire 25.000; Jackson Editrice, edizione 1980 in lingua italiana. Nuovo di zecca, prezzo originale lire 35.000. Pansa Roberto - Via B. Salerno, 28/12 - 16147 Genova - Tel. 010/3993200 (casa) - 293943 (uff.).

Nuovissima TI-58 vendo, mai usata, con confezione completa e modulo base. Cedo anche raccolta di programmi per ingegneria civile per TI-58. Vendo nuovi il Bugbook VI e il Bugbook V. Il tutto a L. 130.000. Vendo anche separatamente. Scrivere a F. Tedesco - Piazza Europa, 15 - 71013 S. Giovanni Rotondo (FG).

AMICOMPUTERCERCO

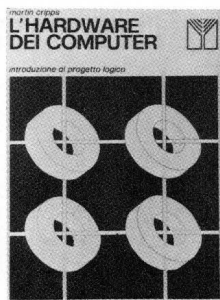
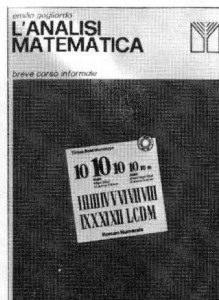
Volete mettervi in contatto con altri appassionati di software e/o hardware? Scrivete a m&p COMPUTER - AMICICOMPUTERCERCO - Via del Casaletto, 380 - 00154 Roma.

Proprietario Computer G.P. desidererei contattare altri interessati per scambio osservazioni, integrazioni e programmi.
Avv. Enrico Abeatici - Via Valdirivo, 23 - 34132 Trieste - Tel. 040/64680.

SERVIZIO LIBRI

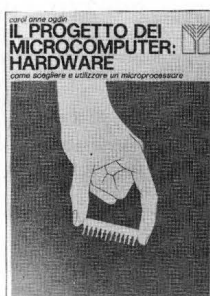
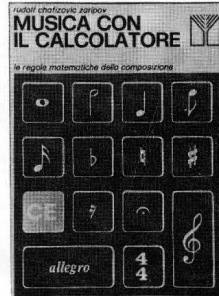
PER RICEVERE CONTRASSEGNO I LIBRI AL VOSTRO INDIRIZZO compilate il tagliando pubblicato qui a fianco in ogni sua parte e speditelo (in busta) a: m&p COMPUTER - Servizio Libri Gruppo Editoriale Suono - Via del Casaletto, 380 - 00151 ROMA

L'HARDWARE DEI COMPUTER

di Martin Cripps
Franco Muzzio Editore
Per i lettori con un buon bagaglio tecnico od elettronico, ci sono ottime opere sui dettagli del progetto dei computer. Tuttavia, gli studenti che intraprendono lo studio dei computer alle scuole superiori o all'università, hanno di solito un bagaglio limitato di nozioni tecnologiche, ed è per essi che il libro è stato scritto, basandosi sugli appunti di corsi tenuti dall'autore presso l'imperial College of Science and Technology di Londra. Il testo è inoltre adatto a coloro che, non interessandosi direttamente di computer, desiderino rimuovere ogni ostacolo che li separa dai misteri delle «scatole colorate con le luci lampeggianti». **Lire 9.500.**

L'ANALISI MATEMATICA

di Emilio Gagliardo
Franco Muzzio Editore
Il libro riporta alla lingua italiana gli esperimenti didattici compiuti dall'autore durante otto anni trascorsi come insegnante nei corsi di Advanced Calculus e Calculus presso Università americane dove in continuo dialogo informale con gli studenti (che amichevolmente obbligano il docente ad essere utile a loro stessi e alla società) ha contribuito a modificare il punto di vista sul significato dell'Analisi Matematica e sul modo di apprenderla. **Lire 9.500.**

IL PROGETTO DEI MICROCOMPUTER: HARDWARE

di Carol Anne Ogdin
Franco Muzzio Editore
Il testo descrive tecniche che l'autore usa ogni giorno e molte altre persone hanno sperimentato con successo. La trattazione è adatta a coloro che non hanno particolari conoscenze.
Passo dopo passo il lettore verrà condotto attraverso le problematiche della progettazione di un sistema a microcomputer, ed imparerà ad affrontare e risolvere le varie situazioni che via via si presenteranno. In questo volume l'accento è posto particolarmente sull'aspetto dell'hardware. **Lire 13.500**

MUSICA CON IL CALCOLATORE

di Rudolf Chafizovic Zaripov
Franco Muzzio Editore
Il libro è dedicato al problema della composizione di musica con l'aiuto di calcoli matematico-probabilistici. Viene esposta una rassegna degli studi svolti in tutto il mondo sull'aiuto che i computer possono fornire per la composizione o per l'analisi della musica. Vengono poi esposte le regole trovate dall'autore per rendere la macchina elettronica capace di realizzare un modello che simila l'attività di un compositore. La monografia contiene circa 400 righi musicali e intende essere utile anche a coloro che effettuano analoghe ricerche sui modelli scientifici di altre attività. **Lire 9.500**

LA COMBINATORIA COMPUTAZIONALE

di A.S. Page e L.B. Wilson
Franco Muzzio Editore.
La combinatoria computazionale è l'aspetto moderno della matematica combinatoria, la disciplina che studia le strutture finite enumerandole, computandole e organizzandole. L'uso del calcolatore permette di affrontare nuovi temi combinatorici, come la complessità degli algoritmi; d'altra parte con la combinatoria si possono affrontare problemi di natura informatica, come l'analisi degli algoritmi. Questo volume mette in luuce i collegamenti tra combinatoria ed informatica e ne analizza gli aspetti fondamentali. **Lire 13.500**

LE SCIENZE CON IL CALCOLATORE TASCABILE

di David R. Green e John Lewis
Franco Muzzio Editore
Tratta, passo dopo passo, le varie funzioni disponibili sui calcolatori e dimostra come si possono applicare a moltissimi tipici problemi di fisica, chimica, biologia, matematica, ingegneria. Vengono introdotti metodi numerici utili agli studenti di scienze e di ingegneria e vengono descritte esasttamente le loro implementazioni sui calcolatori tascabili, riportando le sequenze dei tasti necessari sui due tipi di calcolatori: quelli con logica algebrica e quelli con logica polacca inversa. Vi sono contenuti numerosi esempi svolti e un grande numero di problemi presi dalle scienze, che il lettore deve svolgere. **Lire 11.000**

32 PROGRAMMI PER IL PET

di Tom Rugg e Phil Feldman
Franco Muzzio Editore
Completamente documentati, pronti per essere eseguiti su ogni tipo di PET. Ogni programma si compone di: (1) Scopo: il significato del programma. — (2) Come usarlo: dettagli sul modo di eseguire il programma, sul significato delle possibili opzioni, sulle limitazioni dell'algoritmo. — (3) Esecuzione di prova: fotografie nelle quali si può vedere lo schermo del PET durante una esecuzione-tipo. — (4) Lista del programma. — (5) Semplici variazioni: suggerimenti per cambiare il programma. — (6) Routine principali: elenco e spiegazione. — (7) Variabili principali: elenco e spiegazioni. — (8) Progetti suggeriti: idee per variazioni. Il volume è stato accuratamente adattato per la lingua italiana; tutti i programmi sono stati verificati. Circa 250 pagine, 60 fotografie, 32 liste di programmi. **Lire 9.500**

MATEMATICA CON IL CALCOLATORE TASCABILE

di Peter Henrici
Franco Muzzio Editore
Contiene 35 programmi scritti per l'HP-33E e per l'HP-25. I programmi implementano algoritmi di teoria dei numeri, soluzioni di equazioni, teoria della stabilità algebrica, analisi delle serie di potenze, integrazione e funzioni speciali, come le varie funzioni di Bessel e la funzione zeta di Riemann. I programmi sono completati dai diagrammi di flusso e dalle istruzioni operative. Lo scopo principale del libro consiste nel far sperimentare concretamente le caratteristiche degli algoritmi presentati, discutendone velocità e convergenza, accuratezza e affidabilità. La prima collezione di programmi matematici ad alto livello per un calcolatore programmabile. **Lire 15.500**

SERVIZIO LETTORI

Per ricevere informazioni direttamente dal distributore di un prodotto presentato in questo numero di m&p COMPUTER, compilate uno dei tagliandi pubblicati qui a fianco e SPEDITELO DIRETTAMENTE AL DISTRIBUTORE, in busta o su cartolina postale. Potete raggruppare più richieste sullo stesso tagliando, se rivolte allo stesso distributore. L'indirizzo viene pubblicato vicino al numero di riferimento. Non spedite a m&p COMPUTER; riceverete più rapidamente le risposte, sia perché viene eliminato uno dei viaggi per posta, sia perché vengono facilitate le operazioni di evasione da parte dei distributori, ai quali i tagliandi giungono scaglionati invece che tutti insieme.

m&p COMPUTER SERVIZIO LETTORI

Desidero ricevere informazioni sul prodotto identificato sul n. 8 di m&p COMPUTER dal riferimento servizio lettori

Spedire direttamente al distributore
Mittente (nome e indirizzo):

....
....
....
....

Desidero ricevere informazioni sul prodotto identificato sul n. 8 di m&p COMPUTER dal riferimento servizio lettori

Spedire direttamente al distributore
Mittente (nome e indirizzo):

....
....
....
....

Desidero ricevere informazioni sul prodotto identificato sul n. 8 di m&p COMPUTER dal riferimento servizio lettori

Spedire direttamente al distributore
Mittente (nome e indirizzo):

....
....
....
....

Desidero ricevere informazioni sul prodotto identificato sul n. 8 di m&p COMPUTER dal riferimento servizio lettori

Spedire direttamente al distributore
Mittente (nome e indirizzo):

....
....
....
....

Ricordate di indicare completamente e con chiarezza il vostro indirizzo!

m&p COMPUTER SERVIZIO LIBRI

m&p COMPUTER - Servizio Libri

Desidero ricevere contrassegno all'indirizzo sotto indicato i seguenti libri:

Numero di copie	Titolo	Prezzo unitario	Importo totale
________	L'analisi matematica	Lire 9.500	________
________	Le scienze con il calcolatore tascabile	Lire 11.000	________
________	L'hardware dei computer	Lire 9.500	________
________	Musica con il calcolatore	Lire 9.500	________
________	Matematica con il calcolatore tascabile	Lire 15.500	________
________	Il progetto dei microcomputer: hardware	Lire 13.500	________
________	La combinatoria computazionale	Lire 13.500	________
________	32 programmi per il PET	Lire 9.500	________
		Totale generale	________

(+ spese postali)

____
(Firma)

Cognome ____________ Nome ____________

Indirizzo ____________ N. ____________

C.A.P. ________ Città ____________ Provincia ____________

SERVIZIO LETTORI

Per ricevere informazioni direttamente dal distributore di un prodotto presentato in questo numero di m&p COMPUTER, compilate uno dei tagliandi pubblicati qui a fianco e SPEDITELO DIRETTAMENTE AL DISTRIBUTORE, in busta o su cartolina postale. Potete raggruppare più richieste sullo stesso tagliando, se rivolte allo stesso distributore. L'indirizzo viene pubblicato vicino al numero di riferimento. Non spedite a m&p COMPUTER; riceverete più rapidamente le risposte, sia perché viene eliminato uno dei viaggi per posta, sia perché vengono facilitate le operazioni di evasione da parte dei distributori, ai quali i tagliandi giungono scaglionati invece che tutti insieme.

m&p COMPUTER n. 8 — SERVIZIO LETTORI
Spedire direttamente al distributore

Spazio per eventuali note

....
....
....
....

m&p COMPUTER n. 8 — SERVIZIO LETTORI
Spedire direttamente al distributore

Spazio per eventuali note

....
....
....
....

m&p COMPUTER n. 8 — SERVIZIO LETTORI
Spedire direttamente al distributore

Spazio per eventuali note

....
....
....
....

m&p COMPUTER n. 8 — SERVIZIO LETTORI
Spedire direttamente al distributore

Spazio per eventuali note

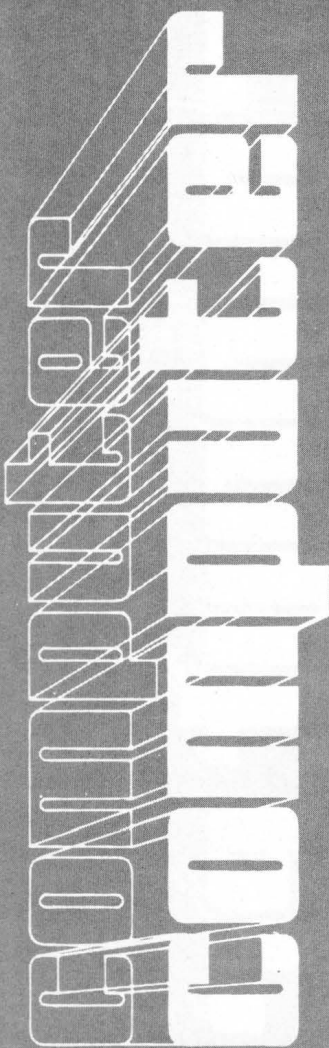
....
....
....
....

m&p COMPUTER SERVIZIO LETTORI

Ricordate di indicare completamente e con chiarezza il vostro indirizzo!

Spedire (in busta) a:
m&p COMPUTER — Servizio Libri
Via del Casaletto, 380
00151 — ROMA

m&p COMPUTER SERVIZIO LIBRI

CAMPAGNA ABBONAMENTI

CAMPAGNA ABBONAMENTI

valida fino al 31 dicembre 1980

12 NUMERI DI m&p COMPUTER A LIRE 22.000
ABBONATEVI A m&p COMPUTER!

Riceverete tutti i numeri, senza pericolo che in edicola vadano esauriti. Risparmierete 8.000 lire rispetto all'acquisto in edicola e sarete al riparo da eventuali aumenti del prezzo di copertina!

Tutti gli abbonamenti sottoscritti con la cedola pubblicata qui sotto hanno *decorrenza dal numero 10*. Solo in pochi casi potremmo riuscire, infatti, ad attivare l'abbonamento fin dal prossimo numero: i conti correnti impiegano almeno 10-15 giorni, ma anche fino a 40-45, per arrivare fino a noi, specie se provengono da piccoli centri; i possibili ritardi delle lettere sono noti a tutti. In questo modo intendiamo evitare che venga acquistato in edicola un numero che poi arriva per abbonamento. Per esigenze organizzative, non sono ammessi abbonamenti con decorrenza da numeri arretrati. Per la richiesta di arretrati è possibile utilizzare la stessa cedola dell'abbonamento. È valida anche una fotocopia della cedola.

m&p COMPUTER
CAMPAGNA ABBONAMENTI

m&p COMPUTER — CAMPAGNA ABBONAMENTI

Desidero sottoscrivere un abbonamento a 12 numeri di m&p COMPUTER, con decorrenza dal numero 10 al prezzo speciale di:

- ☐ L. 22.000 (Italia)
- ☐ L. 26.000 (ESTERO: Europa e Paesi del Bacino Mediterraneo)
- ☐ L. 42.000 (ESTERO: Americhe, Giappone, Asia ecc.: spedizione VIA AEREA)

Lire ______________

Desidero ricevere i seguenti numeri arretrati di m&p COMPUTER

(a L. 3000 ciascuno):.. Lire ______________

TOTALE: Lire ______________

Scelgo la seguente forma di pagamento:

- ☐ Versamento sul c/c postale n. 774018 intestato a Gruppo Editoriale Suono — Via del Casaletto 380 — 00151 ROMA (ricordate di indicare la causale del versamento!)
- ☐ Allego assegno intestato a Gruppo Editoriale Suono
- ☐ Attendo il vostro avviso

Cognome ______________________ Nome ______________________

Indirizzo ______________________ N. ______

C.A.P. ________ Città ______________________ Provincia ______________

Compilate anche questa parte per favore
- ☐ Nuovo abbonato
- ☐ Rinnovo
  si prega di indicare il numero di codice abbonato o allegare la fascetta di spedizione ....

....
(Firma)

INDICE INSERZIONISTI

Spedire (in busta) a:
m&p COMPUTER — Ufficio Diffusione
Via Giovanna Gazzoni, 42
00133 — ROMA

IRET
informatica

The C8000 Series is a compatible family of microcomputer-based systems, designed specifically for business applications.

These powerful general-purpose systems combine processor, memory, fixed 8-inch disk, and cartridge tape drive — all within one low-profile enclosure.

The C8001 is an 8-bit system that's ideal for one or two users. And it's easily upgraded to the more powerful 16-bit C8002 configuration, which can handle up to eight users.

Based on the Z8000* processor, the C8002 can be connected to a high-speed local network for further expansion.

Industry compatible versions of *COBOL, BASIC, FORTRAN* and *Pascal* are available on several operating systems, including an adaptation of the *UNIX** timesharing system. Also available are packages for communications, data base management, word processing and business applications.

Inside or out, We're all business.

Onyx C8000 Series

Distributore esclusivo per l'Italia DATA SYSTEMS

ADVEICO S.r.l. - SEDE LEGALE: Via A. Tadino, 22 - 20124 Milano - Tel. 02/2043281
UFFICI AMMINISTRATIVI E COMMERCIALI: Via Emilia Ovest, 129 - 43016 S. Pancrazio (Parma) - Tel. 0521/998841 (2 linee urbane)